ALGORITHMS, APPLICATIONS AND SYSTEMS TOWARDS INTERPRETABLE PATTERN MINING FROM MULTI-ASPECT DATA

by

Xidao Wen

Submitted to the Graduate Faculty of

the School of Computing and Information in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2019

UNIVERSITY OF PITTSBURGH

SCHOOL OF COMPUTING AND INFORMATION

This dissertation was presented

by

Xidao Wen

It was defended on

October 14th, 2019

and approved by

Dr. Yu-Ru Lin, University of Pittsburgh

Dr. Peter Brusilovsky, University of Pittsburgh

Dr. Konstantinos Pelechrinis, University of Pittsburgh

Dr. Christos Faloutsos, Carnegie Mellon University

Dissertation Director: Dr. Yu-Ru Lin, University of Pittsburgh

ALGORITHMS, APPLICATIONS AND SYSTEMS TOWARDS INTERPRETABLE PATTERN MINING FROM MULTI-ASPECT DATA

Xidao Wen, PhD

University of Pittsburgh, 2019

How do humans move around in the urban space and how do they differ when the city undergoes terrorist attacks? How do users behave in Massive Open Online courses (MOOCs) and how do they differ if some of them achieve certificates while some of them not? What areas in the court elite players, such as Stephen Curry, LeBron James, like to make their shots in the course of the game? How can we uncover the hidden habits that govern our online purchases? Are there unspoken agendas in how different states pass legislation of certain kinds? At the heart of these seemingly unconnected puzzles is this same mystery of multi-aspect mining, i.g., how can we *mine* and *interpret* the hidden pattern from a dataset that simultaneously reveals the associations, or changes of the associations, among various *aspects* of the data (e.g., a shot could be described with three aspects, player, time of the game, and area in the court)? Solving this problem could open gates to a deep understanding of underlying mechanisms for many real-world phenomena. While much of the research in *multi-aspect* mining contribute a broad scope of innovations in the mining part, interpretation of patterns from the perspective of users (or domain experts) is often overlooked. Questions like what do they require for patterns, how good are the patterns, or how to read them, have barely been addressed. Without efficient and effective ways of involving users in the process of multi-aspect mining, the results are likely to lead to something difficult for them to comprehend.

This dissertation proposes the M^3 framework, which consists of **m**ultiplex pattern discovery, **m**ultifaceted pattern evaluation, and **m**ultipurpose pattern presentation, to tackle

the challenges of multi-aspect pattern discovery. Based on this framework, we develop algorithms, applications, and analytic systems to enable interpretable pattern discovery from multi-aspect data. Following the concept of meaningful multiplex pattern discovery, we propose *PairFac* to close the gap between human information needs and naive mining optimization. We demonstrate its effectiveness in the context of impact discovery in the aftermath of urban disasters. We develop *iDisc* to target the crossing of multiplex pattern discovery with multifaceted pattern evaluation. *iDisc* meets the specific information need in understanding multi-level, contrastive behavior patterns. As an example, we use *iDisc* to predict student performance outcomes in Massive Open Online Courses given users' latent behaviors. *FacIt* is an interactive visual analytic system that sits at the intersection of all three components and enables for interpretable, fine-tunable, and scrutinizable pattern discovery from multi-aspect data. We demonstrate each work's significance and implications in its respective problem context. As a whole, this series of studies is an effort to instantiate the M^3 framework and push the field of multi-aspect mining towards a more human-centric process in real-world applications.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

When I started my doctoral study, I was never convinced that I would be able to pull it off. This was largely because the Ph.D. study started right after a major setback in my health. On the one hand, life has been shinning again without pains in my nerves, and on the bright side, this experience has broadened my values of life, where getting a Ph.D. is merely one of them. On the other hand, as someone who stayed in school for most of my student career, I was not sure a Ph.D. study would be as exciting as I had expected. It took a tremendous amount of courage, aspiration, and hard work to complete this doctoral study. For that, I thank myself for never giving up, staying to the end, while taking care of myself.

I also considered myself lucky to have a group of mentors and friends around in this process. I am deeply grateful to my advisor, Professor Dr. Yu-Ru Lin, who has been extremely helpful in the course of my doctoral study. Beyond the incredibly approachable personality, she always has sharp instincts and insights in research and never hesitates to challenge your work and provide instant and constructive suggestions. How she develops, delivers, and exchanges ideas have made exemplars of a responsible, responsive, and continuously reflective {advisor, collaborator, and friend}. I wanted to "complain" though that, Yu-Ru invited us relocating to the most secured room from all the beautiful window offices scattered in the building, where we used to sit. However, this has turned out to be a remarkable changing point since all lab members came closer to each other, and this has triggered more exciting discussions and cultivated a stronger and more supportive community environment of the Picso lab. I wanted to thank Dr. Peter Brusilovsky, who took me in as a Ph.D. student of his PAWS lab in the darkest point of my life, for which I would forever appreciate it. Aside from his infinite number of questions in research meetings and fun characteristics in-person, his course "adaptive information systems" has been one of my favorites in Pitt. In one of his lectures, he asked the students to speak one sentence "the user is not like me"

out aloud and repeat three times. This short sentence, in fact, conveys a lot of lessons in designing any systems that involve humans to operate, and more importantly, this is a lesson of life always to put yourself in other people's shoes. I also wanted to appreciate Dr. Konstantinos Pelechrinis. The collaboration inspires my exploration of Multi-Aspect data mining with Kostas in analyzing the impact of bombing attacks in Boston. This direction of work becomes more concrete and complete with his insightful suggestions and comments, especially in the design of *PairFac* and *FacIt*. Kostas is also someone that I admire, who always pushes the boundary in his well-established fields while having the desire to pursue new areas driven by what he is passionate about. I would also like to thank Dr. Christos Faloutsos, who managed to serve on my committee with all the responsibilities at CMU and on his sabbatical leave. I would always think of Christos as someone who has a big name in the field, carries a big smile, encourages to think big, and provides big support.

I have always enjoyed having Yongsu Ahn, Xian Teng, Muheng Yan, Mengdi Wang, Mert Ertugrul, Wen-Ting Chung and Xingsheng He around to talk about everything in Picso Lab. My friends in PAWS lab, Rosta Farzan, Denis Parra, Claudia López, Sherry Sahebi, Julio Guerra, Yun Huang, and Roya Hosseini, are like my family at Pitt and provided me with tremendous support and love when I was not well. Thanks to my friends: Shuguang Han, Marcin Koźniewski, Martijn de Jongh, Chun-Hua Tsai, Jidapa Kraisangka, and Sung-Min Kim, who enriche my Ph.D experience at Pitt; Marc Meijer, Jesse Lundberg, Michael Cobb, Seth Shelnutt, PJ Santoro, and Katja Wald, with whom I have argued, shared coffee, and drank with at Cambridge; my colleagues, Jie Chen, Ke Zhang, and Velin Konnev for having me at AT&T labs in Bridgewater and Rui Chen, Na Wang, Muheng Xie and Xi Liu for hosting me at Samsung Research America in Sunnyvale.

Last but not least, I would like to thank my family, who have always been supportive and considerate in every aspect of my life. I am truly grateful to have: my parents who set examples of consistency, persistence, dedication and devotion; my sisters who endured my nonsense when I was growing up, and support my dreams throughout my life; my partner, together with whom we experience excitements, achievements, moments that are not for us, and anything that is coming to us.

Xidao Wen, 2019, Pittsburgh

1.0 INTRODUCTION

Given a set of observations that describe spatio-temporal human activities in a city, what are the underlying livehoods [43] that summarize the urban dynamics? How do these phenomena shift when the city experiences disasters such as terrorist attacks and earthquakes? Considering data from a Massive Open Online Course platform, where students can access different material at various time points in a course from a diverse set of mediums (e.g., laptop, iOS), can we uncover their implicit learning habits and how they connect with course-end performance? Considering play-by-play data about shot choices of NBA players, where we know which player makes the shot, the time in the game, and the area on the court, can we understand hidden shot-selection preferences? At the heart of these seemingly isolated problems is the same core of multi-aspect data mining.

Multi-aspect data can be considered as a collection of records each of which is associated with different aspects of information. Many real-world processes and phenomena can be regarded as multi-aspect data, such as the human movements in the urban space, the user behaviors on online platforms, and the play-by-play data in sports that we discuss above. For example, a shot in the NBA data is described with three aspects: player, time in the game, and area on the court. Multi-aspect data mining then seeks to address the following problem: how can we *mine* and *interpret* hidden patterns from a dataset that reveal the associations, or changes of the associations, among various *aspects* of the data. Solving this problem helps us navigate through complex observations of massive size and concisely reveal the underlying mechanisms for many real-world phenomena.

Current research in multi-aspect data mining mostly focuses on mining alone, including sparse [11,126], scalable [2,20,96], or coupled [171,209] multi-aspect data mining applications, contributing to efficient pattern mining from sparse, heterogeneous data sources in different

domains. While the resulting patterns are made available to domain experts, they are not necessarily in accessible or interpretable forms. In fact, the need for interpretation of patterns from the perspective of domain experts is often overlooked. Issues such as what experts require for a pattern to be useful, how good the patterns are, and how to read them pose challenges when mining with multi-aspect data (detailed in Chapter 1.1). Without efficient and effective ways to involve users in the process of multi-aspect mining, the results can be difficult for them to comprehend and apply to their work.

At a broad level, Machine Learning models have demonstrated great success in learning patterns and accurately predicting a wide variety of complex phenomena. As we move from decision trees to multi-layer perceptron, from principle component analysis to autoencoders, as we achieve remarkable milestones on an extensive range of tasks, like neural machine translation and image recognition, we need to ask how well we can explain the models. While this need is related in discussions of interpretable machine learning, most work is limited to the domain of supervised learning [50,142,145,150], providing different taxonomies or practices of interpretability in machine learning. Although these studies present some valuable understanding and guidance, they presume the learning task is a supervised nature. There remains very little discussion about interpretability in unsupervised tasks. There is a need to empower unsupervised learning methods with an understanding of interpretability.

We can certainly try to apply the interpretability methods of supervised settings to unsupervised ones. However, we must carefully evaluate their applicability before implementation. The reason we are uncertain about their direct applicability is that the interpretability of supervised learning tasks focuses on the ability to probe, understand, and trust decision support systems (e.g., forecasting or classifications); however, unsupervised tasks, such as multi-aspect data mining, help explore the previously unknown patterns in the data without explicit labels. While there have been studies that look at issues of interpretability in unsupervised learning (e.g., [28,32,113,205]), they target specific application scenarios. There has not been a comprehensive framework developed for interpretability in unsupervised learning.

This dissertation takes the first initiative to study the interpretability framework for unsupervised learning settings. We are interested in studying mining problems in the context of multi-aspect data because an increasing amount of data are in multi-aspect formats. Multi-

aspect data is often represented as a *Tensor* 1 (formal definition in Chapter 2.1), where the dimensions of the tensor correspond to different aspects of the data. As tensor factorization is one of the most popular methods that takes in multi-aspect data and uncovers a set of underlying data structures (or patterns), this dissertation focuses on the interpretable tensor factorization of multi-aspect data.

1.1 PROBLEM STATEMENT

There are several challenges involved in interpretable unsupervised learning from multi-aspect data, due to the discrepancy between how humans conceptualize data and existing tensor methods and their evaluations. To situate the problem in a concrete context, let us consider the NBA shot analysis data in the 2014-2015 season, as an example of multi-aspect data. Figure 1 (a) shows several tuples of the shot data, which consists of three attributes associated with each shot made: zone in the court, name of the players, and the quarter when the shot was made. We can use a three-dimensional tensor $\mathcal{X}$ ($time \times player \times zone$) to represent such multi-aspect data, where each entry in the tensor denotes the number of shots taken by player p at time t in zone z. Given $\mathcal{X}$, tensor factorization generates a set of R patterns, so that a reconstructed tensor based on these patterns can best resemble the original tensor $\mathcal{X}$. While I will formally define the term "pattern" later on, a pattern from tensor $\mathcal{X}$ is simultaneously explained by the *descriptor* player, *descriptor* zone, and *descriptor* time, indicating a tendency of the shooting pattern over *player*, *zone*, and *time*, respectively (formal definitions of *descriptor* will be given in Chapter 2). In this section, we discuss the challenges in mining from multi-aspect data.

1.1.1 (C1) Mining With Mismatch Between Human Information Need/Interest And Naive Error-Based Optimization

Error-based optimization means that the pattern discovery process is lead by the minimization of the error between the original tensor and reconstructed tensor. Simply applying

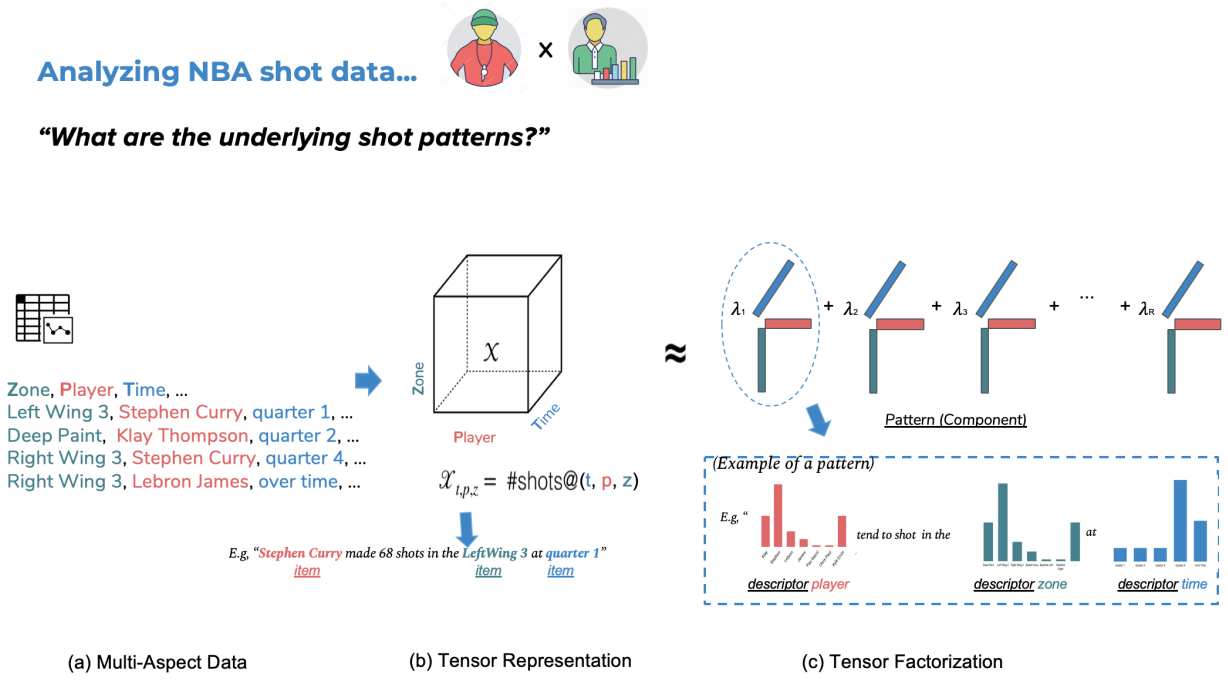

Figure 1: Illustration of using tensor to represent multi-aspect data and applying tensor factorization to simultaneously reveal the associations among aspects of the data.

conventional mining algorithms for multi-aspect data, such as HOSVD [47], CP decomposition [105] or Tucker decomposition [221] could lead to the discovery of latent patterns that have limited meaning to users (readers are referred to Section 2.3.1 of Chapter 2 for detailed discussions). This is because discovery via off-the-shelf methods is the process of optimizing an objective function to minimize the discrepancy between the original tensor and the reconstructed tensor based on the latent patterns. However, it does not consider the nuances of human information need when interpreting the patterns. One such need is that the latent patterns are usually enforced to be non-negative, meaning they must be greater than or equal to zero. This need is reasonable, even mandatory, to represent the probabilistic-like tendency of a set of items, e.g., players, zones, temporal units in our toy example. Consider a three-dimensional tensor of *player* $\times$ *zone* $\times$ *time*. Each of the patterns can be jointly explained by a tendency over players, zones and quarters. In this case, to characterize such

a tendency, we need the patterns to be non-negative (Figure 1 (c)) so that they can have probabilistic interpretations. In addition to non-negativity, there are other common needs for the sake of better interpretability of the latent patterns, such as sparsity and smoothness.

A promising solution is to understand the particular interests of users under specific application scenarios and integrate their requirements as part of optimization goals. For example, users of Tensor Factorization might sacrifice its fit for increased sparseness [11] or non-negativity [192] in the latent representation. However, when human information need is beyond simple additions of characteristics to the patterns, existing model constraints usually can not account for it. One of the goals of this dissertation is to discover underlying patterns that are catered to specific human information needs.

1.1.2 (C2) Mining With Insufficient Evaluation Criteria

In supervised learning, the goal is to predict a response or make a decision. Therefore, part of the job of interpretable machine learning is to explain a response or decision being made. Since there is typically no response or decision involved in unsupervised learning, it is not immediately clear how to systematically evaluate the results of models given the varying selection of evaluation approaches. As we will discuss in detail in Section 2.4 of Chapter 2, there are several practices of evaluation in tensor factorization.

The first practice is to look at the overall *quality* of the reconstruction via proxies, such as reconstruction error [42, 64, 102, 126, 224, 238, 262], root mean square [21, 25, 195, 226], or less often, mean square error [152]. This is an essential start to evaluating the model results as experts need to know the extent to which the results are a faithful representation of the original data. For applications that also concern pattern discovery, experts tend to focus on *validating* the patterns [1, 2, 242]. In this case, interpretable unsupervised learning from multi-aspect data has to explain the underlying data phenomenon in meaningful terms to the domain experts. For example, with the set of shot patterns returned from tensor factorization in Fig.1(c), experts typically follow "I know it when I see it" [66] to examine whether each shot pattern makes sense or not.

On the other hand, unsupervised methods, such as clustering [237], matrix factoriza-

tion [112], and graph representation learning [77] aim to process unlabelled data and output a description of their latent structures. These structures often shed light into underlying relationships within the data and can be useful in downstream tasks. For example, link prediction and entity classification are commonly seen as external tasks to evaluate the performance of embedding algorithms [250]. We expect that knowledge gained from unsupervised methods from multi-aspect data could be used to reveal relationships between underlying data structure and predictive outcomes. More specifically, we try to find what kinds of *utility* we can add given the patterns. For example, can we use the results from the tensor factorization to group players into different shot styles? If so, how are these styles aligned with experts' domain knowledge? There is a line of research that evaluates the mining from the perspective of performance in downstream tasks, such as recommendations [21,33,54,121,122], classifications [102,128,178,261], or clustering [55,260]. However, as we have seen, researches take on different evaluation criteria. We argue that comprehensive guidance of evaluation in multi-aspect mining could make it more likely for research in the field to stimulate experts' understanding of the mining outputs.

1.1.3 (C3) Mining With Mismatch Between Experts' Domain Knowledge And Data-driven Models

In real-world applications, data can be noisy and the solution space for multi-aspect mining can be large due to the non-uniqueness property [130, 181] of the factorization. As a result, a data-driven model with an adequate fit does not necessarily translate to one that experts can justify with their domain knowledge.

A popular approach to mitigate this issue is to incorporate domain-specific knowledge so that the solution space can be reduced to areas that are constrained by the domain knowledge (e.g., [4, 6, 9, 11, 54, 92, 121]). Consider again our toy example of the NBA dataset in Figure 1. One piece of our experts' domain knowledge could be that Klay Thompson shots the ball in a very similar way to Stephen Curry. In this case, the knowledge can be used as auxiliary information in the tensor factorization to constrain the similarity between players. As a result, the resulting solution space can be geared more towards an area confined by

experts' domain knowledge.

Unfortunately, existing studies presume such knowledge to be available in advance and in a structured format, e.g., matrix or graph. This can be impractical because such auxiliary information (such as pair-wise relationships between players): 1) can be difficult/expensive to acquire; 2) is subject to change betweem interpretations of different experts (e.g., expert A can see it differently than expert B); and 3) can be implicit (e.g., Klay Thompson is closer to Stephen Curry than Lebron James is) rather than explicit (hierarchical structure in the spatial dimension, e.g., state→city→county). Furthermore, when most approaches present the latent patterns to experts, the experts have no way to incorporate their knowledge flexibly. Another focus of this dissertation is to provide a human-in-the-loop solution that allows users to interact with tensor models to steer the discovery interactively.

1.1.4 (C4) Mining With Mismatch Between The Underlying Multi-Aspect Model Complexity And Human Understandability

Like topic modeling [22], or matrix factorization [112], the results of unsupervised discovery from multi-aspect data often do not readily translate to how humans see things clustered or close to one another. As we observe from Figure 1 (c), to interpret one pattern, we need to simultaneously look at three descriptors (player, zone and item) before knowing that the pattern suggests a clustering of how players tend to shoot the ball in a heat-map of zones in the court and with a tendency over different quarters of the game. The complexity increases dramatically with the rise of the dimension of the tensor and the number of patterns. Therefore, the challenge is to maintain human understanbility as multi-aspect models' complexity increases.

A plausible remedy could be to present patterns in a way that they adapt to how humans tend to interpret them, connecting the data to information [72]. However, there has been little work to understand the specifics of human information needs for experts to easily understand patterns from multi-aspect data. What exactly an interpretable data phenomenon denotes is unclear. Therefore, it is also the focus of this dissertation to understand the set of requirements of experts to interpret the latent patterns and propose ways to address their

needs.

1.2 RESEARCH QUESTIONS

Based on the above problem statements, this dissertation is designed to address the following three research questions:

1.2.1 RQ1: How can we conduct pattern discovery with human information need?

My first research question aims to close the gap between human information need and error-based optimization (**C1**). Existing work has provided ways of addressing certain information needs, such as non-negativity, sparsity, and smoothness, which can be easily solved via well-known constraints. However, when the information need is beyond enforcing a characteristic on the patterns, existing models are often unable to reconcile this. Therefore, we would like to understand the specific information needs and devise models that can incorporate them into the pattern discovery process.

1.2.2 RQ2: How can we design a rigorous evaluation schema for the factorization results?

Once we have identified the information need and proposed solutions of pattern discovery tailored to this specific information need, my second research question aims to understand how we can evaluate the effectiveness of the results (**C2**). Given various evaluation strategies in literature, we would like to propose rigorous evaluation guidance so that the results are vetted from multiple, complementary perspectives.

1.2.3 RQ3: How can we keep the experts in the loop of pattern discovery?

The above studies present pattern discovery as a one-shot approach, where experts are given the outputs and left to agree or disagree after evaluating the quality of the results. However,

the ultimate goal of pattern discovery is to make patterns meaningful to domain experts. To facilitate this process, experts need to quickly understand the results. There should be some mechanisms to solicit feedback from the experts so that the patterns can be adjusted based on experts' inputs. Therefore, there are two key questions we need to address:

- RQ3a: How can we facilitate experts' understanding of the results from tensor factorization? (**C4**)
- RQ3b: How can we allow the tensor factorization to take and weigh in experts' feedback on the results? (**C3**)

1.3 RESEARCH FRAMEWORK

Based on the above examination of the challenges in mining with multi-aspect data, we propose the goal of this dissertation as follows:

The goal of this dissertation is to provide a framework that users can follow for the design of interpretable unsupervised learning from multi-aspect data, by answering research questions **RQ1-3** that address the challenges **C1-4**.

In this dissertation, we propose the M^3 framework for interpretable unsupervised learning, which consists of three components that tackle each of the research questions: **Multiplex Pattern Discovery** (→**RQ1**), **Multifaceted Pattern Evaluation** (→**RQ2**), and **Multipurpose Pattern Presentation** (→**RQ3**).

As a first step, this thesis aims to close the gap between human information needs and error-based optimization by devising models through the process of **Multiplex Pattern Discovery** from multi-aspect data, given users; exploration purpose. Multiplex pattern discovery requires multiplex objective functions that simultaneously consider human information needs and naive error-based optimization, by understanding the human information needs and translating them to specific optimization objectives. We present a case study of multiplex pattern discovering in the context of event analytics. The information need is

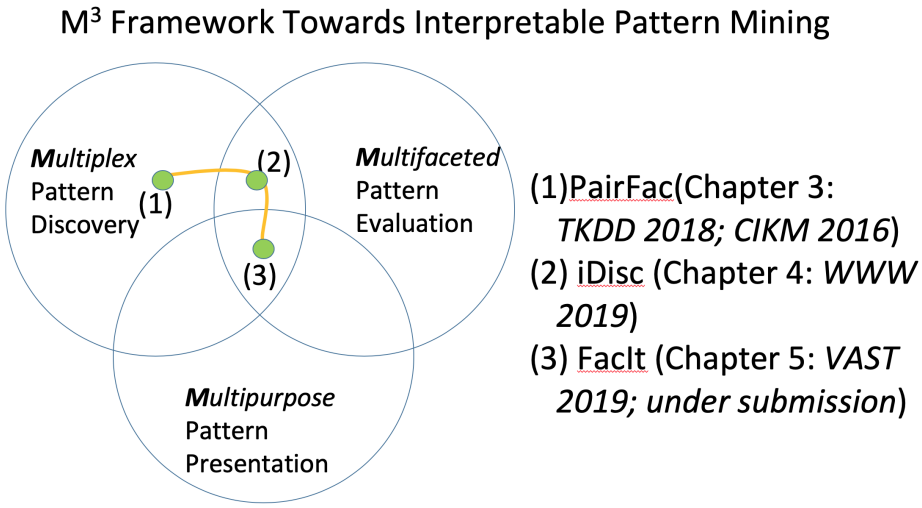

Figure 2: **Research Framework of this dissertation**. The thesis proposes the M^3 framework for interpretable tensor factorization from multi-aspect data. The framework consists of three components: multiplex pattern discovery, multifaceted pattern evaluation and multipurpose pattern presentation. Chapter 3 solves the need for multiplex pattern discovery, by devising *PairFac* to close the gap between human information need and tensor reconstructions in the context of impact discovery in the aftermath of urban disasters. Chapter 4 targets the intersection of multiplex pattern discovery with multifaceted pattern evaluation, using *iDisc* to meet the specific information need to understand user behavior patterns in MOOCs while connecting their behaviors to performance outcomes. Chapter 5 sits at the intersection of all three components as it introduces a visual analytic system, *FacIt*, for interpretable, tunable and scrutinizible pattern discovery from multi-aspect data.

understanding the changes in multi-aspect mobility data before and after major events in a city. While human information needs go far beyond the above, we use event analytics as case study because the nature of the task (cross-examination of patterns from a pair of multi-aspect data) shares generic information needs with many applications in different domains, such as pattern comparisons in normal and anomaly multi-aspect time-series analysis, in benign and malignant bio-maker discovery, etc. We design a collective tensor factorization model, *PairFac* (Chapter 3), to uncover shared and discriminative phenomena given the multi-aspect data being split into two groups, before and after an event. In this work, we target a typical type of multi-aspect data, human mobility data. Our proposed method will be able to answer the question: how does the event change *when*, *where*, and *what* citizens normally do in a city? *PairFac* formulates this pattern discovery problem as a discriminant tensor analysis problem and solves it through joint factorization of a *pair* of tensors. Given two tensors capturing urban activity data *before* and *after* a potentially impactful event, *PairFac* learns the shared and discriminative latent subspaces of the tensor pairs, while revealing which patterns persist and change across multiple aspects of urban activity data. Our comprehensive experiments of *PairFac* on both synthetic and real-world event datasets demonstrate its effectiveness.

Second, we establish the utility of **Multiplex Pattern Discovery** and **Multifaceted Pattern Evaluation** in the context of understanding the association between latent multi-aspect user behavioral phenomena and performance outcomes in MOOC platforms. When comparing data structures of users from different performance groups, differences can reside either in high-level or fine-grained patterns. Revealing patterns' hierarchical structure would add value to the understanding of semantic relationships among the patterns. To this end, we propose a tensor-based learning method— iterative Discriminative tensor factorization, *iDisc* (Chapter 4) —that discovers the common and discriminative learning patterns at multiple levels. In addition to a typical process of involving experts to qualitatively validate the patterns, we introduce the utility examination of the patterns, connecting them to students' course-end performance outcomes. Empirical studies with datasets from different MOOC platforms have shown promising results of the effectiveness and efficiency of *iDisc*.

The last study of this dissertation takes a step back, aiming to develop a unified system

that addresses interpretability in the process of **Multiplex Pattern Discovery**, **Multi-faceted Pattern Evaluation**, and **Multipurpose Pattern Presentation** in a general unsupervised pattern mining setting from multi-aspect data. We introduce *FacIt* (Chapter 5), a generic visual analytic system that directly factorizes tensor-formatted data into a visual representation of patterns to facilitate result interpretation, scrutinization, information query, and model selection and refinement. After interviewing our domain experts, we propose a design that incorporates (i) a suite of model scrutinization and inspection tools that allow efficient tensor model selection, (ii) an interactive visualization design that empowers users with characteristics- and content-driven pattern discovery, and (iii) a novel weakly-supervised tensor factorization algorithm to support human-in-the-loop model adjustment. Based on this multipurpose pattern presentation, *FacIt* solicits human information needs with a set of novel pattern interaction mechanisms and incorporates user feedback and input into the factorization process. *FacIt* also follows a more rigorous evaluation schema; results are vetted based on their quality, validity and utility. We demonstrate the effectiveness of our system through usage scenarios across different domains and in-depth expert interviews.

1.4 OVERVIEW OF THE CHAPTER STRUCTURE

The structure of this dissertation is as follows.

- Chapter 2 surveys related works and aims to position this dissertation in the literature.
- Chapter 3 presents the collective tensor factorization for pattern discovery from multi-aspect data.
- Chapter 4 introduces an iterative pattern discovery framework with the goal of revealing shared patterns and discriminative patterns at multiple levels and their utility in finding associations between performance outcomes and user behaviors on MOOC platforms.
- Chapter 5 describes a human-in-the-loop visual analytic framework for pattern discovery that allows users to incorporate experts' feedback in the process of pattern discovery.
- Chapter 6 summaries the dissertation, acknowledges its limitations, and lays out directions for future work.

2.0 BACKGROUND

In this chapter, we provide the preliminaries of tensor factorization, followed by a comprehensive set of surveys in the area of mining with multi-aspect data. The first two sections provide an essential understanding of tensor terminologies (Section 2.1) and multi-aspect data phenomena (Section 2.2) for readers to connect with their own background. The remaining sections look at existing work in multi-aspect data mining from the perspectives of mining models (Section 2.3), evaluation strategies (Section 2.4), and presentation forms (Section 2.5), respectively.

2.1 TENSOR PRELIMINARIES

In this section, we provide the essential background to understand the basics of tensors, algorithms commonly used in discovery patterns from tensors, and existing applications of tensors given the diverse nature of multi-aspect data.

2.1.1 Tensor Basics

In this section, we provide the preliminaries to understand tensors, many of which follow the conventions provided [105]. Table 1 presents the notation we use in the rest of the paper.

Tensor. A tensor is a mathematical representation of a multi-aspect data array, i.e., an extension of concepts such as scalars, vectors and matrices to higher dimensions. We use x to represent a scalar, $\mathbf{x}$ a vector, $\mathbf{X}$ a matrix, and $\boldsymbol{\mathcal{X}}$ a tensor.

Table 1: Description of Notations Used in this Dissertation.

Symbol	Description
x	a scalar (lower-case letter)
$\mathbf{x}$	a vector (boldface lower-case letter)
$\mathbf{X}$	a matrix (boldface capital letter)
$\boldsymbol{\mathcal{X}}$	a tensor (boldface Euler script letter)
$\mathbf{X}_{i,j}$	the scalar at the $\{i, j\}$ position of matrix $\mathbf{X}$
$\boldsymbol{\mathcal{X}}_{\boldsymbol{i,j,k,...}}$	the scalar at the $\{i, j, k...\}$ position of $\boldsymbol{\mathcal{X}}$
$\mathbf{X}_{(m)}$	mode-m unfolding of tensor $\boldsymbol{\mathcal{X}}$
$\mathbf{U}^{(m)}$	mode-m factor matrix of tensor $\boldsymbol{\mathcal{X}}$
$\mathbf{U}_r^{(m)}$	the r-th column in mode-m factor matrix of tensor $\boldsymbol{\mathcal{X}}$
I_1,..., I_M	the dimensionality of mode 1, ..., M
R	the desired rank (Capital Italic script letter)

Indexing We further use $\mathbf{x}_i$ to denote the i-th entry of vector $\mathbf{x}$, $\mathbf{X}_{ij}$ to denote the element of matrix $\mathbf{X}$ at position $\{i, j\}$ and $\boldsymbol{\mathcal{X}}_{ijk}$ to denote the element of third-order tensor $\boldsymbol{\mathcal{X}}$ at position $\{i, j, k\}$. The *order* of a tensor is the number of dimensions (also referred to as modes, or ways).

Order. The *Order* of a tensor is the number dimensions used to represent the multi-aspect data. Therefore, a scalar x can be considered as zero-order tensor, vector $\mathbf{x}$ a first-order tensor, matrix $\mathbf{X}$ a second-order tensor, and $\boldsymbol{\mathcal{X}}$ being a third-order or high-order tensor. The order of a tensor can also be referred to as its way, i.g., third-order tensor can also be referred as a three-way tensor.

Mode. We also use *mode* or facet to denote each specific aspect of the multi-way tensor. The first mode, or second mode is another way to refer the first or second dimension in the tensor data. The *dimensionality* of a mode is the number of elements in that mode. We use I_q to denote the dimensionality of the q-th mode. For example, the three-way tensor $\boldsymbol{\mathcal{X}}$

$\in \mathbb{R}_{+}^{I_1 \times I_2 \times I_3}$ has three modes with dimensionality of I_1, I_2, and I_3, respectively. $\mathbb{R}_{+}$ indicates that all the elements of $\boldsymbol{\mathcal{X}}$ obtain non-negative values.

Rank. In the tensor decomposition we introduced later, R denotes the specified *rank* – the number of components. The problem of determining the rank of a given tensor is NP-hard [81]. In practice, the rank is determined numerically by fitting various rank-R model for $R = 1, 2, \ldots$ until a "good" model is found. However, we argue that the numerical goodness of fit should not be the only criterion for specifying the rank. Other criteria such as model compactness and interpretability are also important for a tensor decomposition results to be practically useful. In this work, we tackle the problem of the selection of rank as a part of our task.

2.1.2 Basic Operations

Vectorization. Vectorization is the process of converting multi-aspect data to an unidimensional array. Given an M-way tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$, vectorization transforms $\mathcal{X}$ to a vector $\mathbf{x} \in \mathbb{R}^{\prod_i^M I_i}$.

Matricization. Matricization is the process of reordering the elements of a multi-way array into a matrix. A mode-n matricization of a M-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ is denoted by $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times \prod_{q \neq n}^{M} I_q}$.

Mode-n product. The mode-n matrix product of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ with a matrix $\mathbf{U} \in \mathbb{R}^{J \times I_n}$ is denoted by $\boldsymbol{\mathcal{X}} \times_n U$ and is a new tensor of size $I_1 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$ with $(\boldsymbol{\mathcal{X}} \times_n \mathbf{U})_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 i_2 \cdots i_N} u_{j i_n}$.

Khatri–Rao. Khatri–Rao product between two matrices $\mathbf{A} \in \mathbb{R}^{I \times K}$ and $\mathbf{B} \in \mathbb{R}^{J \times K}$, results a matrix $\mathbf{C} = \mathbf{A} \odot \mathbf{B}$, where $\in \mathbb{R}^{(IJ) \times K} \in \mathbb{R}^{(IJ) \times K}$ corresponds to the column-wise Kronecker product.

2.1.3 Tensor Factorization Algorithms

This section introduces the most commonly used tensor decomposition techniques—CP decomposition and Tucker decomposition.

CP Decomposition. CANDECOMP/PARAFAC [80] decomposition is often referred to as CP decomposition. The CP decomposition of tensor $\boldsymbol{\mathcal{X}}$ could be expressed as $\boldsymbol{\mathcal{X}}_{opq} \approx \sum_{r=1}^{R} \mathbf{A}_{or}\mathbf{B}_{pr}\mathbf{C}_{qr}$. Let $[\mathbf{z}]$ denote a superdiagonal tensor, where $[\cdot]$ is the operation that transforms vector $\mathbf{z}$ to a superdiagonal tensor by setting tensor element $z_{k\ldots k} = \mathbf{z}_k$ and other elements as 0. Thus the CP decomposition of a three-way tensor can be written as $\boldsymbol{\mathcal{X}} \approx [\mathbf{z}] \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$. Following Kolda [107], the CP model can be concisely expressed as $\boldsymbol{\mathcal{X}} \approx [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \equiv \sum_{r=1}^{R} \mathbf{A}_r \circ \mathbf{B}_r \circ \mathbf{C}_r$. Non-negative Tensor Factorization can be considered a special case of CP decomposition, where we wish to find a non-negative rank-R tensor to approximate the original tensor $\mathcal{X}$.

Tucker Decomposition. Tucker decomposition decomposes a tensor into a smaller/core tensor multiplied by a matrix along each mode. For the case of a three-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, we have $\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{Z}} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$. Matrices $\mathbf{A} \in \mathbb{R}^{I_1 \times O}$, $\mathbf{B} \in \mathbb{R}^{I_2 \times P}$, and $\mathbf{C} \in \mathbb{R}^{I_3 \times Q}$ are called *factor matrices*, or *factors/components*, while tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is called the *core tensor*. In this process, each element of the tensor $\boldsymbol{\mathcal{X}}$ is the product of the corresponding factor matrix elements multiplied by a weight $\boldsymbol{\mathcal{Z}}_{opq}$, i.e., $\boldsymbol{\mathcal{X}}_{i_1 i_2 i_3} \approx \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} \boldsymbol{\mathcal{Z}}_{opq}\mathbf{A}_{oi_1}\mathbf{B}_{pi_2}\mathbf{C}_{qi_3}$.

2.1.4 Tensor Factorization Results

We introduce key terms that we use throughout the dissertation to describe the typical outputs of tensor factorization: descriptors, factor matrices, and patterns. As presenting and understanding the results requires precise notation of the patterns, we introduce the descriptor as foundation of the building blocks: descriptors from multiple modes are used to jointly describe a pattern and the descriptors which correspond to each mode can be combined to describe the factor matrix.

Descriptor. We refer to each entry i for $i = 1, \ldots, I_m$, as an *item* of the m-th mode in the tensor. We denote the vector $\mathbf{u}_r^{(m)} \in \mathbb{R}^{I_m}$ as a *descriptor* consisting of entries $\langle \mathbf{u}_{ir}^{(m)} \rangle$ for $i = 1, \ldots, I_m$ from the m-th mode that describes the contribution of the i-th *item* i to the r-th component. For a non-negative tensor, it is often useful to constrain the descriptor to take non-negative values to facilitate the interpretability of occurrence-likelihood, i.e., $\mathbf{u}_r^{(m)} \in \mathbb{R}_+^{I_m}$

where an entry value $\mathbf{u}_{ir}^{(m)}$ can be considered how likely the i-th item is associated with the r-th component. Example of descriptors are shown in Figure 1. They include players, zones, and quarters.

Factor Matrix. Factor matrices refer to the combination of the vectors from the rank-one components. Given an M-way tensor, the mode-m factor matrix $\mathbf{U}^{(m)}$ is a collection of descriptors $\mathbf{u}_r^{(m)}$, i.g., $\mathbf{U}^{(m)} = \left[\mathbf{u}_1^{(m)^T} | \mathbf{u}_2^{(m)^T} | \dots | \mathbf{u}_R^{(m)^T}\right] \in \mathbb{R}^{I_m \times R}$. For example, for the three-way tensor in Figure 1, tensor factorization results in three factor matrices that correspond to players, zones, and quarters, respectively. A factor matrix can also be considered the latent embeddings for the respective mode obtained via tensor factorization, where each item i in the m-mode is given a vector representation of $\mathbf{u}_i^{(m)} = \left[\mathbf{u}_{i1}^{(m)}, \mathbf{u}_{i2}^{(m)}, \dots, \mathbf{u}_{iR}^{(m)}\right] \in \mathbb{R}^R$.

Pattern. The r-th component or *pattern* is a collection of descriptors from each mode $C_r = \{\mathbf{u}_r^{(1)}, \dots, \mathbf{u}_r^{(M)}\}$. In this work, "pattern" and "component" are used interchangeably while "pattern" also refers to a component as a visual representation. In Figure 1, the interpretation of each pattern needs to examine all three of its descriptors associated.

2.2 MULTI-ASPECT DATA PHENOMENA

Massive multi-aspect datasets have emerged from many fields. In this section, we review the different ways that multi-aspect data can be structured. The existing applications of multi-aspect data are typically categorized by domain, e.g., social networks, healthcare, chemistry, computer vision, etc. The following schema looks at the nature of the data and classifies them by the objects that the data has been collected around.

2.2.1 Multi-Aspect Data that Describes Singular Objects

One type of multi-aspect data describes various sets of variables related to a set of objects. In this case, we could have different sets of variables measured on different samples, e.g., different conditions or times, where objects can be any meaningful entities or research interests, such as process batches($batch \times time \times variables$ [156,157]), physical locations ($sites \times time \times$

indicators [16,58,115,148,179,206]), patients (*patient* × *medication* × *diagnosis* [86,87,229]), users in a social network (*nodes* × *time* × *measurements* [158, 168]), authors in text-based systems (*author* × *time* × *keyword* [106, 207, 208]).

In process control, Nomikos et al. [156] use multi-aspect data to monitor batch processes. Each batch is associated with a set of measurements, including flow rates of styrene, the temperatures of the feeds, the reactor, and the density of the latex in the reactor at a sequential interval of 5 minutes. In environmental research, Lee et al. [115] construct a multi-aspect dataset to represent hourly indoor air quality index measurements for various sites, e.g., NO, NO_2, NO_x, CO and PM2.5. In the healthcare domain, authors [87] work with the data of patients' diagnoses and their corresponding procedures to derive phenotypes. In social networks, one dimension could be the actors in the network and the other dimensions could be different types of measurements related to the actors. For example, Oliveira and Gama [158] build a tensorial representation of the student network by measuring the degree, eigen centrality, closeness, and betweenness centrality of each student at different snapshots of time to track the evolution of dynamic social networks. In text-based systems, Sun et al. [207] extract a three-order tensor from DBLP data to encode authors' keywords in their publications for each year.

Regardless of the distinct domains, multi-aspect data can be used to characterize a certain type of entities using longitudinal or cross-sectional measurements. In the case where longitudinal measurments are taken, the multi-aspect data can be also regarded as multivariate data.

2.2.2 Multi-Aspect Data that Describes Pairwise Objects

The second category of multi-aspect data records the measurements related to two sets of objects. A simple case would be a multivariate image that presents various wavelenths as variables for pixels, which have x-coordinates and y-coordinates, therefore having *row* × *column* × *measurements* [116, 117, 183, 217, 253, 256]. The goal of such a data construction is to discover the relationships between the objects in the cross-category. Researchers have used this scheme to construct corresponding multi-aspect datasets in various other domains, e.g.,

network security ($originIP \times destinationIP \times time$ [12, 135, 137]), transportation ($origin \times destination \times time$ [98, 214, 215, 226]), and social networks ($person \times person \times time$ [170, 177, 239]).

With this multi-aspect representation, hyper-spectral images can be simply represented as third-order tensors: two ways for rows and columns and one way for the spectral band [183]. The signal subspace that integrates the spatial and spectral information has lead to significant improvement in target detection. Video data has also been represented as a 3D tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, where I and J are the spatial dimensions of a video frame and K is the total number of frames [217]. Designating two ways to represent the same set of entities bears more expressiveness in their interactions. In domain of network security, Maruhashi et al. [137] builds a four-way tensor from the port number and the time ticks of the network traffic from the source IP to the destination IP. Leveraging a tensor-based representation of the heterogeneous traffic network enables the discovery of structured relationships. The same ideas are also often applied to social networks and transportation networks. Peng and Li [177] construct a three-way tensor from the email exchanges between 184 users in 44 months, based on the Enron email dataset. Want et al. segment the Beijing area and build a three-way tensor based on taxi trips between 651 zones in 24 hours to understand the spatial-temporal structure of the traffic dynamics. Each of the element in the tensor indicates the volume of traffic from the i-th origin area to the j-th destination area in the k-th time domain.

We have also seen multi-relational data [154] structured in the form of tensorial representation. In this scenario, the multi-aspect data represent the dyadic relational data which consist of n entities and m relations. One example of such is the knowledge graph [23], where different entities in the graph are linked by various types of relations. Modeling as multi-aspect is an effective, straightforward solution for multiple binary relations [154]. High-dimensional, sparse spaces are a generic setting where factorization models achieve competitive results [153, 219].

Most of the work surveyed builds tensors to describe pairwise objects in the tensor, rather than a singular object. Mining from such representations can decouple the latent embeddings of the entities into separate parts as the objects are examined with meaningful

semantics. In the transportation data, the areas can have a representation that indicates where people tend to depart and another representation of where people tend to arrive. In relational data, the entity can have dedicated representations that treat them as subjects in addition to objects. Additional modes, such as time and measurements, are used to describe the varying natures of the relations between these entities under different contexts. For example, the traffic in the morning is different from that in the evening.

2.3 MULTI-ASPECT DATA MINING

Multi-aspect data mining has been applied in a wide range of domains. Despite the diversity of domains, this section surveys the work that is categorized in two classes, classic multi-aspect data mining as straightforward applications of tensor factorization and knowledge-driven data mining as results of data fusion.

2.3.1 Multi-Aspect Data Mining

Classic pattern mining techniques from multi-aspect data employ vanilla tensor factorization models to decompose the data into a set of latent patterns. One line of this work is termed tensor completion, which focuses on recovering or de-noising the tensor [44,63,109,127,201,213,245,255]. For example, Tan et al. [213] use 3D tensors to represent a façade image, natural image and MRI image. After randomly removing the entries from the tensor, the authors are able to apply tensor factorization to recover the missing entries. In medical questionnaires, patients may opt to leave questions unanswered, which could lead to biased parameter estimations. Dauwels et al. [44] have demonstrated that CP decomposition can effectively recover the missing entries in the questionnaires. Rather than searching for meaningful patterns, tensor completion uses the latent components as intermediate steps to reconstruct the tensor. Huang et al. [91] further provide a flexible and efficient framework to impose the above constraints on matrix and tensor factorization for non-negativity and sparsity. Besides, smoothness constraints have played a vital role in recovering missing values in

applications of non-negative matrix factorization with the presence of noise, including blind source separation [247], visual parts extraction [244], and brain signal analysis [31], Yokota et al. [245] impose sparsity constraints when recovering the missing values in the tensor.

Another line of work focuses on using tensor factorization to extract meaningful latent structure from multi-aspect data. The goal is to examine the latent structures to discover hidden insights, where many domains appreciate non-negativity [13,56,57,189,192] for the sake of better interpretability. Bader et al. [13] extract emails from a subset of the Enron email dataset and create an $m \times n \times p$ term-author-month tensor. By using non-negative tensor factorization, the authors are able to assess term-to-author associations both semantically and temporally. Fan et al. [57] propose *CitySpectrum*, an application that uses non-negative tensor factorization to decompose a traffic flow by day, by hour, and by region into basic life patterns given a big mobile phone GPS log dataset. By comparing the patterns before and after the Great East Japan Earthquake, authors are able to understand the fluctuation in people flow due to the earthquake. Espín-Noboa et al. [56] use non-negative tensor factorization to identify clusters of mobility behavior based on taxi trip data in New York City. Although non-negativity often yields meaningful results, additional sparsity may be desired to improve the interpretability of the factors [39,84,87,126,146,159,200,217]. For example, Mørup et al. [146] impose sparsity constraints to reduce ambiguities in the latent components from wavelet transformed electroencephalographic (EEG) data, helping in component identification.

These studies are based on the standard non-negative tensor factorization (NTF) or extensions of it with additional constraint terms such as sparseness, orthogonality, and smoothness. Because of this, the extracted factors can be in any arbitrary form which users have no control over. These approaches have the advantages of being simple and straightforward, since users do not need to supply the observed tensor with other data. However, users have no means to use their domain knowledge to guide the process of factorization.

2.3.2 Multi-Aspect Data Fusion

Chi and Zhu [33] argue that users' prior knowledge of the domain can benefit mining of multi-aspect data in multiple ways. Cognitively, prior knowledge can steer the direction of factorization towards certain subspaces so that are more aligned to human concepts [55]. Statistically, users' input can help alleviate the over-fitting problem [30]. From the application point of view, if users have certain domain knowledge, e.g., a pre-defined ontology [229] or distance metrics [65] of concepts, they can discover through the lens of these priors.

Researchers have proposed various approaches to leverage auxiliary information as domain knowledge to supplement tensor factorization. Following the same idea as relation regularized matrix factorization [118], Narita et al. [152] use an auxiliary matrix to constrain the within-mode similarity. Specifically, given a 3D tensor, authors use the Graph Laplacians [141] inferred by the three similarity matrices to force pairs of objects in each mode to behave similarly to how they are in the spectral space. The approach of leveraging auxiliary information has been shown to improve accuracy in tensor completion tasks. A similar idea is also adopted by Ge et al. [65]. Given a $location \times time \times hashtag$ tensor with random missing values, the authors use external knowledge to construct similarity matrices. They integrate the Graph Laplacians from these within-mode relationships into a tensor completion framework, *AirCP*. Experiments have shown that *AirCP* finds high-quality models of meme spread with access to as low as 20% of the data. Aside from extracting hidden relationships based on Graph Laplacians, Bhargava et al. [21] directly use similarity matrices inferred from the auxiliary data source to the tensor factorization framework in their POI recommendation task.

Bhargava et al.'s work [21] can also be placed in the broad domain of data fusion. In contrast to the other similarity based approaches above, auxiliary information is directly fused into the tensor factorization without the pre-processing step (e.g., extracting the Graph Lapalacians). According to Lahat's defition, data fusion is the analysis of several datasets such that different datasets can interact and inform each other [111]. In data fusion, multiple datasets can be jointly factorized by means of a coupled decomposition of several matrices and/or tensors, with factors shared to varying extents. One of the commonly used frame-

work is Coupled Matrix Tensor Factorization (*CMTF*) [4] proposed by Acar et al., which fuses several data sources and enhances knowledge discovery. Given a set of heterogeneous datasets, Acar et al. propose an all-at-once algorithm to recover missing values in the tensor. The tensor is coupled with multiple matrices, and the tensor and matrices share the same set of factors. Experiments have shown that *CMTF* yields remarkably higher accuracy than CP decomposition when the missing ratio increases. Lin et al. [121] propose *MetaFac* to generalize coupling schemes among heterogeneous data sources. *MetaFac* uses a graph to represent the relations between different data sources, where the nodes represent different data sources and edges denote the shared modes between two data sources.

Both *CMTF* and *MetaFac* requires the sharing of the factors to be complete. However, data from different sources often shares some of the components, but not all. Given the presence of both shared and unshared factors, Liu et al. [128] design a novel factorization algorithm for Common and Discriminative Subspace Non-negative Tensor Factorization(*CDNTF*). *CDNTF* takes a set of labelled tensors as input and computes both their common and discriminative subspaces simultaneously as output. However, this approach requires the prior knowledge of a proper split between the number of shared and unshared factors. Acar et al. [6] further propose a data fusion model that automatically reveals shared and unshared components through modeling constraints. The idea follows *CMTF* with common factors between heterogenous data sources. However, it assigns a sparse weight score to each factor indicating whether it is shared or not. Experiments have shown that this method provides promising results in identifying shared and unshared chemical components.

Despite its success in leveraging additional data to guide in tensor factorization, the framework requires information to be available in advance, which is not possible in many applications. In addition, the framework presents a set of static outputs. Domain users have no way to provide feedback to the model.

2.3.3 Summary

In this section, we reviewed mining methods for multi-aspect data. Vanilla tensor factorization models are quick and convenient tools for users to start with. As users impose more

requirements on the solution space, constraints, such as non-negativity, sparsity, smoothness and orthogonality can be introduced to cater users' different information needs. However, these information needs are generic and preliminary. When the information need is complex, existing model constraints are not able to account.

On the other hand, data fusion models have been used in the presence of heterogeneous data sources as domain knowledge. While various frameworks have shown success in their respective application scenarios, they assume the knowledge is explicit and given prior to factorization. As such knowledge can either be difficult to acquire or implicit in practice, there has not been much work to investigate alternative ways to leverage domain knowledge in this situation.

2.4 EVALUATION IN MULTI-ASPECT DATA MINING

While multi-aspect data phenomena and mining methods have been extensively studied, their evaluation has not been properly vetted. In fact, given the different purpose for mining, different evaluations are often undertaken. There has not been a systematic way to consider the evaluation problem. For the evaluation of tensor factorization, we categorize existing practices in three aspects: quality, validity and utility.

2.4.1 Multi-Aspect Mining Quality

Most of the multi-Aspect mining tasks evaluate in terms of how well the underlying structure resembles the original data. Regardless of varying nature of the mining tasks, the relative reconstruction error (*RRE*) is one of the most commonly used metrics to determine the quality of the factorization [42, 64, 102, 126, 224, 238, 262]. In this line of work, *RRE* has been the key index for determining the rank parameter as well as comparing to alternative methods. Metrics that bear similar meanings are also adopted, such as root mean square area (*RMSE*) [21, 25, 195, 226], mean square error (*MSE*) [152], or *Fit* [260, 261]. In some of the image completion tasks, we have also seen the use of more domain-specific metrics that

are equivalent to the relative reconstruction error, such as signal to distortion (*SDR*) [245], peak signal-to-noise ratio (*PSNR*) [243] and signal-to-interference ratio (*SIR*) [38].

It is important to note that due to the visual nature of the images, it is also common to carry out the qualitative examination between the benchmark images and reconstructed the images, in addition to reporting the quality index. For example, in their work of estimating missing values of visual data, Liu et al. [127] first report *MSE* in comparing with their baseline methods at different parameter settings. Then the reconstructed image is compared with the original image. Similarly, Zhou et al. [260] first presents the comparison of *Fit* measures and then provides the recovered images in comparison with the original ones. A similar practice can also be seen in [37, 245, 261].

2.4.2 Multi-Aspect Mining Validity

A single quality index could be sufficient in most of tensor completion tasks, as the successful recovery of the missing values is the center of attention. As for the applications that require the examination of the resultant underlying structures, the evaluation practice usually varies. Simulated studies [1, 42, 59, 119, 126, 243] are a popular practice in the domain of signal processing, where researchers generate synthetic datasets with ground truth. Then, the results obtained from the mining algorithms would be used to compare with the ground truth.

In these studies, authors usually report the overall factorization quality index and then present a detailed analysis of the performance in recovering the ground truth factors, with the exact order of the two varies, though. In their work of discovering the mobility patterns from traffic data, Want et al. [226] first report the *RMSE* in comparison to baseline methods with varying degree of missing data used in training. Then, the authors provide the empirical study to show insights about the Spatio-temporal structures that are revealed by their method. Takeuchi et al. [212] propose Non-negative Multiple Tensor Factorization (*NMTF*), which factorizes the target tensor and auxiliary tensors simultaneously. In their experiments, they have shown *NMTF* reconstruct better on the synthetic data, measured by reconstruction error. From the online review data sets, they have also demonstrated that *NMTF* can

successfully extract Spatio-temporal patterns of people's daily livelihood patterns.

There are several pieces of work opt-out of reporting the overall factorization quality index, focusing on examining the factor matrices learned from proposed respective models [1, 2, 242]. For example, Cichocki et al. [40] evaluate their work of Hierarchical ALS algorithms to solve non-negative matrix and tensor factorization with a synthetic dataset, comparing the ground truth factors and obtained factors. Other than qualitative examination, we have also seen research taking a more quantitative approach in evaluating the recover quality. In addition to providing the qualitative analysis, Acar et al. [2] also provide a success indicator when the recovered factor matrices resemble the ground truth factors (computed based on cosine similarity $> 99\%$). To evaluate the proposed algorithm in revealing the shared and unshared factors, Acar et al. [1] generate a third-order tensor coupled with a matrix, based on different types of underlying components, sharing different levels of correlation. The resultant factors are compared with ground truth factors, both qualitatively and quantitatively, to confirm the effectiveness of the model.

In the case where ground truth data is not available, there are studies also directly apply the mode on the real-world data sets and involve the experts in qualitatively examining and interpreting the factors [178]. Phan and Cichocki [178] propose a local ALS algorithm that estimates sequentially non-negative components from real-world EEG data containing. The proposed model can find meaningful components that explain the data. With non-negative tensor factorization, Espín-Noboa et al. [56] are able to reveal seven clusters of mobility patterns based on taxi trip data in New York City. Similarly, Bader et al. [13] qualitatively analyze the topics discovered from the Enron email datasets with non-negative tensor factorization. Another recent study [133] also targets at revealing the urban dynamics through decomposing the number of visitors' arrivals in different areas in the city. Based on human-inflow tensor, authors can uncover various urban mobility patterns, such as commuting, leisure, returning home patterns.

2.4.3 Multi-Aspect Mining Utility

There is one line of work interested in evaluating the utility of their models via the applications in downstream tasks [21, 33, 97, 121, 122, 175, 182, 185].

Recommendation tasks [21, 33, 54, 121, 122] are one of the popular evaluation scenarios in the area of social computing. Chi and Zhu [33] propose *FacetCube* for non-negative tensor factorization with prior knowledge. Working on a $author \times keyword \times reference$ tensor, *FacetCube* is evaluated in a recommendation task of top-k queries to most relevant references to authors. To demonstrate the utility of *MetaFac*, Lin et al. [121, 122] conduct prediction tasks asking how effective *MetaFac* can be used to predict what stories a user would "digg" and what stories a user would comment on. In a POI recommendation application, Bhargava et al. [21] also report the recommendation performance. In all of these studies, ranking metrics are commonly used, such as *NDCG* and $precision@k$.

Classification and clustering tasks are also employed in several studies [55, 102, 128, 178, 260, 261]. In these studies, multi-aspect mining is used to extract and select statistically significant (dominant, leading) features that differentiate different classes or clusters. For example, Liu et al. [128] evaluate the effectiveness of *CDNTF* by analyzing its performance in solving a classification problem. In his work, the graph is transformed into a vector representation based on the subspace discovered by *CDNTF*, and then support vector machines (SVMs) are used to build the classifiers to separate different types of chemical compounds as well as atoms. Zhou et al. also evaluate their model in the image classification task [260] and in the image clustering task [261], based on the features extracted from the images.

2.4.4 Summary

In this section, we review the existing practice of evaluation in multi-aspect mining. We have seen that studies have undertaken different types of experimentation strategies. Mining metrics disclose the overall quality of the models, and they are reported across different studies. Examining the factor matrices further validate the patterns as the results of their methods. Qualitative approaches that examine the uncovered patterns are more commonly seen, al-

Table 2: Existing Evaluation Schema in Multi-Aspect Mining.

Studies	Quality	Validity	Utility
[64, 102, 195, 224, 238, 262], [25, 37, 38, 127, 152, 245, 260, 261]	✓	×	×
[1, 2, 59, 119, 212, 242], [13, 40, 56, 133, 178], *PairFac*	×	✓	×
[33, 97, 121, 122, 175, 185]	×	×	✓
[42, 126, 175, 226, 243]	✓	✓	×
[21]	✓	×	✓
iDisc	×	✓	✓
FacIt	✓	✓	✓

though systematic and quantitative comparisons also exist. Utility value is also appreciated, where their models work as feature extraction tools and experiments are conducted through downstream tasks. The rule of thumb in evaluation based on this survey is to report the overall quality index. However, with all the varieties in evaluations, we have seen, there has not been a study using a more systematic and comprehensive evaluation strategy.

2.5 MULTI-ASPECT PATTERN PRESENTATION

The key to interpretability is to *present* the results in understandable terms. This section surveys the existing work for various ways to present the results of the multi-aspect mining. We divide the literature into two categories: ones that statically present the results as in most of the publications; and ones that provide interactive support of pattern presentations as mostly seen in the field of visualizations.

2.5.1 Pattern Presentation in Literature

There are different ways of communicating the multi-aspect patterns to the readers in the existing work. We call it a static pattern presentation because there is no indication of an interactive system being developed to aid their discovery process in their papers. Therefore, we assume the authors of these studies have gone through a process to present selective results as a showcase of the models. We also assume that they explore and interpret all the patterns in the same way that they do in the paper. Since pattern presentation from multi-aspect data usually involves several graphics, one for each descriptor, we categorize their displays into two groups based on how authors have arranged the presentations of different descriptors: adjacent, and isolated alignments.

Few studies examine the results more from the perspective of individual factor matrices. Rather than going through the entire pattern, they look at the columns of each descriptor separately [29, 64, 172, 251, 257]. For example, Chen et al. [29] analyze the speed patterns via tensor decomposition based on a network traffic speed dataset. The authors show the results with each factor matrix instead of organizing the results by patterns. This makes sense as the purpose of the work is to identify interpretable traffic patterns with varying levels of missing values. With such an organization, authors can display how the resultant factor matrix varies with different training data used. However, it can be challenging to comprehend a single pattern as the visual explanation of one pattern is split into various figures. Similarly, Gauvin et al. [64] present the results organized by the factor matrix, instead of the pattern. This makes it easier to see the differences between different components in each factor matrix. However, the trade-off is that it cannot explain the pattern as a whole.

We have seen more often authors use adjacent alignments of descriptors (e.g., [5, 13, 56, 57, 61, 168, 190, 211, 233]). In practice, authors use a dedicated graphic to describe each of the descriptors and then graphics of all descriptors associated with a pattern are positioned side-by-side, to deliver a comprehensive set of perspectives of the pattern. In this way, each graphic provides a complementary explanation of the pattern, and its interpretation involves walking through each figure to get a complete understanding. For example, Williams et al. [233] propose TCA (Tensor Component Analysis) to discover latent components from

three-dimensional tensor of $neuron \times temporal \times trial$ based on a simulated neuron activity dataset. To demonstrates the results, they show eight noteworthy components from a 15-component model, where each of them consists of three graphics from left to right, for the neuron, temporal, and across-trial descriptors, respectively. In another example, Fan et al. [57] proposes *citySpectrum* to model the city dynamics with a four-dimensional tensor $hour \times day \times region \times POI$, based on a mobile GPS dataset. Similarly, they present each descriptor with a dedicated figure and show two interesting patterns related to "entertaining" and "commercial" life patterns in their results.

2.5.2 Interactive Pattern Discovery

More recently, researchers have concerned about the interpretability of pattern discovery from multi-aspect data and attempted to ease the process using visual analytic systems [26, 125,240]. Viola [26] is among the few efforts to interactively present the patterns for anomaly detection in the traffic data. It is a novel tensor-based anomaly analysis algorithm with visualization and interaction design that can dynamically produce interpretable data summaries and allows the domain experts to ranking anomalous patterns. Compared to the existing practice in visualizing results from multi-aspect mining, Viola introduces the interactive pattern exploration mechanism.

TPFlow by Liu et al. [125] uses a piece-wise rank-one tensor decomposition algorithm to automatically slice the data into homogeneous partitions and extract the latent patterns in each partition. Compared to Viola, TPFlow has the advantage of understanding the entire pattern space as a result of the progressive partitioning framework. Yan et al. [240] provide a visual analytic system for pattern discovery in bike-sharing data, which introduces the pattern relation view to describe the relations between the patterns. The pattern relation view is helpful for users to browse patterns quickly and find interesting patterns.

2.5.3 Summary

The existing practice in pattern presentation often hinders the interpretation of the results of the tensor factorization because the pattern presentations are not typically matched with how

Table 3: Existing Work in Pattern Presentation From Multi-Aspect Mining.

Studies	Interactive	Pattern Presentation
[5, 13, 56, 194, 211, 233], [57, 61, 168, 190, 212], [14, 108, 175, 204, 226, 259], *PairFac*, *iDisc*	×	adjacent
[29, 64, 172, 251, 257]	×	isolated
[26, 125, 240]	✓	adjacent
FacIt	✓	multi-scaled, integrated, adjacent

a human perceives they are. The adjacency alignment of the descriptors can be considered as a brute force way of throwing everything about patterns to the domain experts without providing aids in how to connect different descriptors and how to connect different patterns. This problem exaggerates, especially when the number of patterns experts need to go through is large. We argue that to mitigate the mismatch, we need solutions that have "user-first" principles in mind.

We have seen recent efforts in developing a people-centric design of pattern exploration from multi-aspect data [26, 125, 240]. However, they are situated in a spatial-temporal context, which provides a limited understanding of domain experts' requirements when working with and interpreting patterns in a generic multi-aspect data setting. Another aspect of bridging human understandability with pattern presentation is enabling people to be part of the pattern discovery and exploration process. Although Yan et al. [240] allow the users to perform a set of operations with the patterns (e.g., merge, reset, etc.), they are restricted as one-way interaction as the underlying modeling process does not take such feedback into consideration of pattern updating.

3.0 PAIRFAC: EVENT ANALYTICS THROUGH DISCRIMINANT TENSOR FACTORIZATION

This chapter aims to close the gap between human information needs and error-based optimization by devising models through the process of **Multiplex Pattern Discovery** from multi-aspect data. *Multiplex pattern discovery is the idea of devising multiplex objective functions for pattern discovery that simultaneously considers the human information needs and naive error-based optimization, through properly understanding the human information needs and translating them to specific optimization objectives.* This chapter presents the multiplex pattern discover in the problem context of event analytics, where the particular information need is understanding how have the city changed before and after certain major events in multi-aspect mobility. We are particularly interested in multi-aspect mining of event analytics because the nature of such task is the cross-examination of patterns from a pair of multi-aspect data, which shares the generic information needs in different domains, such as pattern comparisons in normal and anomaly multi-aspect time-series analysis, in benign and malignant bio-maker discovery, etc.

3.1 INTRODUCTION

Analyzing the impact of disastrous events has been central to understanding and responding to crises. Effective crisis management requires not only careful planning and preparation for disaster relief operations, but also a timely assessment of an event's impact. The latter is important for facilitating actions that will bring the society back to its normal operations as fast as possible [140]. In this work, we introduce a novel event analysis framework that

can automatically reveal the changes in human behavioral patterns associated with an event through mining context-rich, high-dimensional and potentially heterogeneous urban activity data.

Traditionally, the assessment of a (natural or artificial) disaster's impact has primarily relied on the manual administration and analysis of surveys and questionnaires, as well as the review of authority reports [191]. Both of these approaches are costly and time-consuming. Today, in the era of mobile and pervasive computing, rich digital human traces of routine transactions are generated by city-dwellers, businesses, and organizations that can be collected through online platforms (e.g., activities on social media), sensing technologies (e.g., mobile phones and wireless sensors) and other means (e.g., crowdsourcing platforms). These rich troves of human behavioral data provide an unprecedented opportunity to closely examine - both qualitatively and quantitatively - the changes in urban activity that follow events of interest (e.g., disasters). While much progress has been made in predictive event analytics, such as detecting and/or forecasting event outbreaks [8, 222, 228], automatically quantifying and capturing the impact of an event has been neglected despite its aforementioned importance.

Our objective in this work is to develop an automated method for understanding the impacts of major effects in the urban environment. To achieve our goal we design unsupervised learning techniques to uncover the changes in human mobility patterns before and after an event of interest. In particular, we formulate our objective as a problem of identifying common and discriminative subspaces between two datasets, the first one capturing the behavior of interest prior to the event and the second one capturing the behavior after the event. While there is literature on discriminant subspace learning [75, 101, 128], these solutions fall into the same generic framework that requires the split of shared and discriminative components before learning the subspaces. However, in the context of analyzing the impact of an event, this is not possible. The vast spectrum of disastrous events and the associated context under which they happen, make it extremely difficult to obtain this knowledge. Thus, most of the prior methods cannot be practically applied to disaster event analysis.

In this article, we introduce a novel approach that is able to automatically discover the

impact of an exogenous event. While we are focusing on the impact of an event on urban mobility, our proposed method is generic and can be used to analyze multiple aspects of urban human activities. Our focus on the mobility will enable us to answer the question of how does the event change *when*, *where* and *what* citizens normally do in a city? To reiterate, our approach, called *PairFac*, formulates the problem as a discriminant tensor analysis problem and solves it through the joint factorization of a *pair* of tensors. Specifically, given two tensors capturing urban activity data *before* and *after* an event of interest, *PairFac* simultaneously learns the shared and discriminative latent subspaces of the tensor pairs. *PairFac* thus, reveals the patterns that both persist and change across multiple aspects of urban activity data.

The motivation for designing *PairFac* stems from the fact that understanding the impact of a disastrous event is a necessity in disaster management, while existing methods for discriminative subspace learning exhibit practical limitations in their applicability in reality. More specifically, in the context of disaster management, "impact assessment" plays a critical role in understanding the (social) consequences of an event. In this situation, "social impact" refers to the consequences an event has on human populations, altering the ways in which they live, work, entertain, relate to one another, organize to meet their needs, and cope as members of society [223]. The process of social impact assessment involves a number of steps, including among others "description of the relevant human environment and zones of influence", "identification and investigation of probable impacts", and "estimation of secondary and cumulative impacts". These tasks are traditionally performed through manual, labor-intensive data collection, and comparison. For example, to describe the human environment and zones of influence, relevant data related to the event should be collected and reviewed through a baseline study or community profile. This approach has been limited in terms of scope and comprehensiveness, as it is not possible to scale to all potentially affected people, and is restricted by pre-defined assessment indices that do not necessarily universally apply. Therefore, a data-driven, generalizable, approach that can leverage the large volume of (detailed) data collected from various sources is needed and has the potential to revolutionize the traditional disaster impact assessment process.

Furthermore, existing approaches in analyzing events mostly focus on the impact discov-

ery using one- or two- dimensional analysis (e.g., call activities volume changes [15], change in geographical location distributions [202], change in emotions [230]) and few are capable of discovering multi-dimensional (or multi-modal) impacts. This inevitably leads to significant loss of information associated with certain aspects that are either projected to a lower dimensionality that can be handled by the model used or eliminated all together. Moreover, the interplay between these multiple facets is not explicitly considered and could result in a false interpretation of the outcome. By formulating the problem event impact analysis as a tensor factorization problem, we are able to discover the changes that are correlated in multiple dimensions. For example, the change in the call volumes on top of any association in time can also vary depending on the location of these phone calls (e.g., their distance from the epicenter of the event).

Our method differs from existing work in discriminative subspace learning [75, 101, 128] by introducing the discriminative weight vector that allows for automatically aligning the common components while at the same time discerning the discriminative components. As shown in Fig. 3, we model the mobility data with two three-dimensional tensors[1], one describing the mobility before the event and one describing the behavior after the event of interest. As alluded to above, *PairFac* jointly factorizes the two tensors to identify the latent mobility patterns that both change, as well as persist, before and after the event of interest. Our comprehensive evaluations of *PairFac* on both synthetic and real-world event datasets clearly showcase its effectiveness.

The key contribution of this work includes:

- We formally introduce the problem of capturing the impact of an exogenous event on the normal operations of a system using discriminant tensor analysis. Given the multidimensional nature of the rich human behavioral data, we use tensor representation to preserve the interactions between different information layers, such as the temporal, spatial and human action layers (see Fig. 3).
- We propose *PairFac* (see Fig. 3), a novel joint tensor factorization framework that aims at simultaneously learning the shared and discriminative components from a pair of

[1] *PairFac* can be extended to more dimensions. However, for illustrative purposes, as well as, due to the nature of our datasets, we design and evaluate our method with three-dimensional tensors.

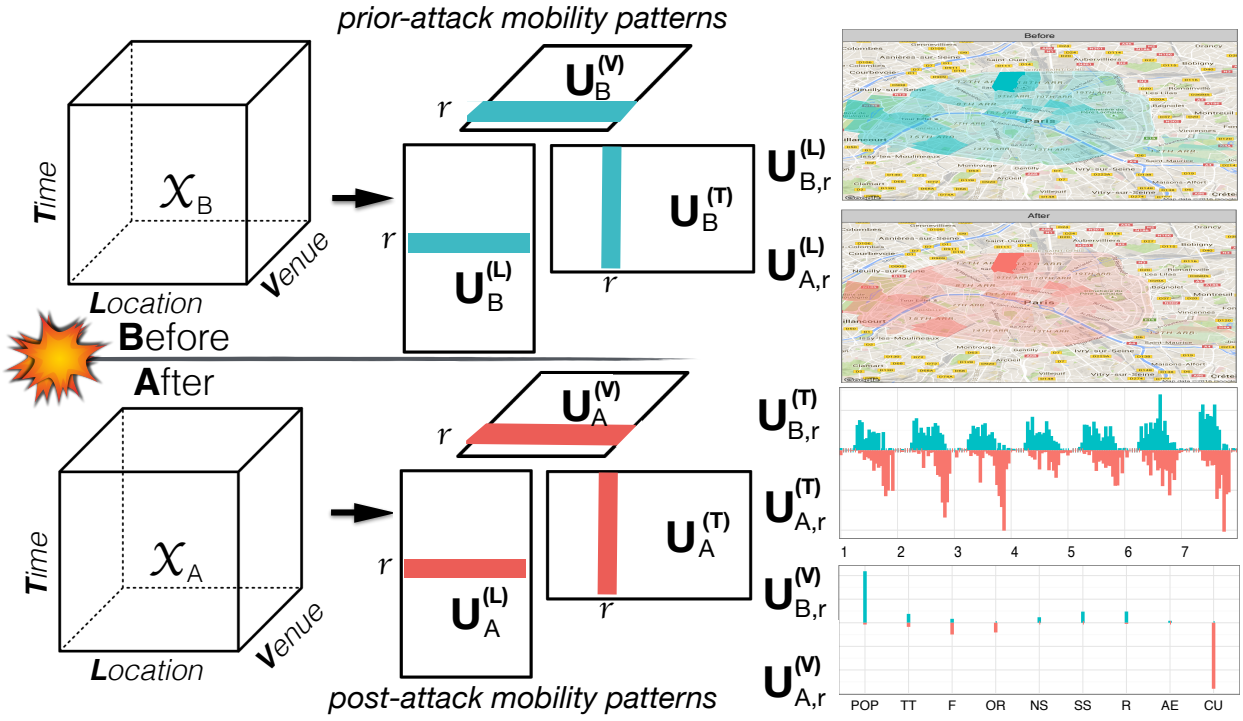

Figure 3: Problem illustration of *PairFac*. $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$ represents the data tensor ***B****efore* and ***A****fter* a specific event (Paris terrorist attack). Matrices $\mathbf{U}^{(L)}$, $\mathbf{U}^{(T)}$, and $\mathbf{U}^{(V)}$ represent the three factor matrices for ***L****ocation*, ***T****ime*, and ***V****enue*, respectively. The same-index columns in each triplet of factor matrix jointly represents a behavioral pattern. *PairFac* identifies similar and discriminative patterns before and after the event. For each pattern (e.g, colored in blue or red), we show the location distribution (e.g., $\mathbf{U}^{(L)}_{B,r}$, $\mathbf{U}^{(L)}_{A,r}$) in the city (of Paris), the time distribution (e.g., $\mathbf{U}^{(T)}_{B,r}$, $\mathbf{U}^{(T)}_{A,r}$) in a week (24×7) and the venue distribution (e.g., $\mathbf{U}^{(V)}_{B,r}$, $\mathbf{U}^{(V)}_{A,r}$) among different activities (e.g., Professional & Other Places (POP), Travel & Transportation (TT), Food (F); please refer to 6.1.2 for details.)

high-dimensional data sources. Our method can automatically identify the common components and at the same time discover the discriminative ones without a predefined number of either type, by formalizing an appropriate optimization problem.

- We provide an efficient iterative algorithm that guarantees convergence to a locally optimal solution for the aforementioned optimization problem. Furthermore, the algorithm is scalable with time complexity linear in the number of non-zero tensor elements. In addition, we provide guidance on a parallel implementation of the algorithm based on Spark that can further speed up the optimization.

This article represents a significant extension of our prior work [231] and is our first full discussion on this subject. In this article, we include new solutions, algorithmic details, and proofs, as well as extensive experimental results. In particular, there are several major developments since our previous work [231]:

- We introduce a new algorithm that provides better interpretation of the discriminative weights while at the same time achieving component alignment. In particular, we introduce an additional auxiliary function to capture the commonalities between the pair of tensors. This additional information gives rise to an easier interpretation of the discriminative scores - i.e., higher scores represent unique patterns while lower scores indicate shared patterns. In addition, our prior work relies on a post-hoc analysis of the learned components to determine the pair-wise alignment of the common components. We address this limitation by re-formulating our objective function with a new regularization term to enforce the similarity between common components.
- We provide a detailed algorithmic description and analysis in addition to a parallelized version of the algorithm. More specifically, we provide details for the solution of our formulated optimization problem, while at the same time providing a theoretical analysis of its convergence. In addition, we provide a scalable, distributed implementation of *PairFac* that speeds up the runtime, through the partition of tensors into mutually non-overlapped blocks. This allows the gradient update in each step to be computed via multiple nodes.
- We perform comprehensive experiments on the scalability and sensitivity of *PairFac*. We

also apply *PairFac* to extensive case studies on real events. To better understand how to appropriately apply the algorithms to event analytics in practice, we systematically analyze the separability of data with respect to the ability of *PairFac* to segment the components into common and discriminative parts. We further employ our approach to discover the long-term impact of terrorist attacks in Paris using traffic sensor data and Twitter geo-tagged content. Another case study is conducted for discovering the changes in mobility patterns during the Thanksgiving week between 2014 and 2015. We demonstrate that our approach can not only distill the crowd activity patterns under exogenous shocks but also analyze long-term activity changes.

The rest of this chapter is organized as follows. Chapter 2.2 discusses literature related to our study, while Chapter 2.3 presents the problem formulation and the essential background. In Chapter 2.4, We introduce multiple solutions to the tensor factorization problem, including a novel algorithm that automatically learns the discriminative weights of the components. Chapter 2.5 provides detailed quantitative results on synthetic datasets, while Chapter 2.6 presents our case studies. In Chapter 2.7 we discuss some open issues and future directions, while Chapter 2.8 concludes our work.

3.2 RELATED WORK

In this section, we describe literature relevant to our methodology and to event and urban analytics.

3.2.1 Shared and Discriminative Subspace Learning

The increasing availability of data from a diverse set of sources has given rise to the study of joint analysis of heterogeneous data. Our study closely relates to the area of discriminative tensor analysis. For example, GTDA (General Tensor Discriminant Analysis) [216] attempts to discover the discriminative features of a pair of tensors as a preprocessing step for subsequent topic discovery and classification tasks, while TCCA (Tensor Canonical Correlation

Analysis) [131] generalizes Canonical Correlation Analysis (CCA) to handle the data of an arbitrary number of views or distinct feature sets and identifies a reliable common subspace shared by all views. Compared to these studies, *PairFac* attempts to simultaneously identify both common and discriminative subspace from the multi-dimensional dataset. The simultaneous discovery of common and discriminative subspace is not new. However, it is typically limited to two dimensions at most. For instance, Gupta et al. [74] propose a joint NNMF on two data sources through a shared subspace, while maintaining their unique variations through individual subspaces. While Gupta et al. [75] further impose mutually orthogonal regularizations to separate the common and discriminative subspaces to ensure that the shared and the discriminative subspaces are mutually exclusive. Following the same idea, Kim et al. [101] relax the framework by requiring the shared subspaces to be *similar* while not necessarily being strictly identical. Regarding the shared and discriminative subspace learning in the context of tensor factorization, the framework by Liu et al. [128] - similar to [74] - separates the subspace into shared and individual subspaces. In contrast, *PairFac* imposes regularization on the shared and discriminative subspaces to automatically identify the number of either type of components, while offering scalability by enabling factorization of even higher dimensional data.

3.2.2 Event Analytics

During the last few years, there has been an increasing interest in the area of event analytics through microblogs (e.g., Twitter). Researchers have approached this field from three perspectives. One line of research is geared towards large-scale societal event detection and forecasting (e.g., civil unrest, disease outbreak, and elections). A common technique is to monitor the frequency of all words and look for a sudden burst in the frequency of (a subset of) them [138]. For instance, Ning et al. [155] develop a multiple instance learning based approach to identify evidence-based precursors and forecast events into the future.

The second line of research aims at sense making of an event's storyline through statistical analysis or visual analytics. For example, Diakopoulos et al. [48] design a visual analytics tool to help journalists and media professionals extract news-worthy content from a large

volume of social media data.

Our work falls into the third line of research, which aims at studying the impact of an event on the affected population. E.g., Lin and Margolin [120] explore the emotional response of Twitter users in different cities to the bombing attacks in Boston, while, Bagrow et al. [15] provide a quantitative view of the behavioral changes in human activity under extreme (natural and man-made) conditions, such as bomb attacks and earthquakes, through the analysis of mobile phone records. In a similar direction, Song et al. [202] mined GPS traces of 1.6 million users and built a system to automatically discover, analyze, and simulate the mobility of a large population under severe disasters in Japan. The shortcoming of using only cell phone and GPS data is that the activity context is absent. Including information relevant to activity concept significantly complicates the analysis due to the increased dimensionality of the data.

3.2.3 Urban Computing

In recent years, there has been a significant volume of research in the area of urban computing and informatics. Zheng et al. [258] summarize seven types of urban computing scenarios for urban planning, transportation, environment, energy, social issues, economy, and public safety and security. Our work, from an application perspective, falls into the last category, as we seek to obtain an understanding of the impact of exogenous events on urban space. Recently, there have been several inspiring studies looking at urban environments. For instance, Pan et al. [164] detect traffic anomalies based on drivers' routing behavior on road networks, while, Pang et al. [165] apply the likelihood ratio test (LRT) (widely used in epidemiological studies) to describe traffic patterns. Our research is geared more towards the area of disaster analytics in urban environments. Early forecasting and detection of disasters are critical for planning effective humanitarian interventions and disaster management. However, there is plenty of literature in this realm, and it is not the focus of our study. For example, Lee and Sumiya [114] propose to detect events such as environmental disasters from geo-tagged Twitter data, while Yu et al. [246] propose multiple Markov boundaries in local causal discovery to identify the sets of precursors for tornado forecasting. In another study,

Madaio [132] developed the *Firebird* framework to help municipal fire departments identify and prioritize commercial property fire inspections, with a combination of techniques from machine learning, geocoding and information visualization. Finally, the short- and long-term evacuation plans/behaviors in the case of a disaster have also been studied [202, 203].

The contribution of our work resides in the area of disaster impact discovery from multidimensional and heterogeneous data. *PairFac* is a generic framework that can be used to study the impact of various exogenous events – being either natural, man-made, or imposed by the local government (e.g., planning policies). For example, the impact of a long-term construction project on the inhabitants' mobility and activities can be quantified using *PairFac*. Identifying behavioral changes for a variety of "urban interventions" has been identified as an open problem pertaining particularly to urban computing [258].

3.3 PROBLEM FORMULATION

3.3.1 Problem Formulation

Simultaneous Discovery of Common and Discriminative Activity Patterns:

Problem Definition 1. Let us consider two non-negative tensors, $\boldsymbol{\mathcal{X}}_B \in \mathbb{R}^{I_L \times I_T \times I_V}$ and $\boldsymbol{\mathcal{X}}_A \in \mathbb{R}^{I_L \times I_T \times I_V}$ representing the urban activities ***B****efore* (B) and ***A****fter* (A) an exogenous shock, where the tensor modes represents the ***L****ocation* (L), ***T****ime* (T) and ***V****enue* (V) of the activities. We seek to obtain a non-negative tensor factorization (NTF) to approximate both input tensors, as $\boldsymbol{\mathcal{X}}_q \approx [\![\mathbf{U}_q^{(L)}, \mathbf{U}_q^{(T)}, \mathbf{U}_q^{(V)}]\!], \forall q \in \{A, B\}$, and $\mathbf{U}_q^{(m)} \in \mathbb{R}_+^{I_m \times R}, \forall m \in \{L, T, V\}$, represents the factor matrices corresponding to each mode.

Note, as alluded to above, that in this work we focus on three-mode tensors but *PairFac* can be used to deal with data with higher dimensionality. The location dimension corresponds to specific neighborhoods in the city, the time is quantized hourly, while the venue dimension captures the various types of establishments available (e.g., coffee shops, retail shops, etc.). The term "venue" refers to the kind of place people visited, e.g., restaurants, schools, etc. – that is, the semantics of the human activity, whereas "location" refers

to the geographical location that certain activity occurs. In our data, "venue" information is available from Foursquare that describes the functionality or the activity provided by the point of interest (or location). As shown in Fig.3, the corresponding columns (red) of each factor matrix together define a mobility pattern that associates specific areas/neighborhoods, time, and types of venues. Disastrous events, such as terrorist attacks, can inject intensive psychological instabilities in the targeted population and as a result the mobility and/or behavioral patterns of this population are likely to change after the event. The goal of the problem described above is to discover the shared and discriminative components of the tensor structures describing their urban activities before and after an event of interest.

3.4 SOLUTIONS

In this section, we begin with providing solutions to Problem 1. We start by describing the current state-of-the-art approaches to solving similar problems [75, 101, 128] (Sections 4.1 and 4.2). We then discuss their limitations and introduce a new solution (Section 4.3). We further provide a parallel implementation of our solution in Section 4.4.

3.4.1 Shared and Discriminative Subspace Approach

To learn the shared and discriminative subspace, Liu et al. [128] proposed the **C**ommon and **D**iscriminative subspace **N**on-negative **T**ensor **F**actorization (CDNTF) which takes a set of tensors as its input and computes both their common and discriminative subspaces simultaneously as the output. Following their work, the objective of CDNTF can be re-written as the following simultaneous factorization of two input tensors: $\boldsymbol{\mathcal{X}}_q \approx [\![\mathbf{U}_q]\!]$, where $\mathbf{U}_q = [(\mathbf{U}^{(m)}_{q:C}, \mathbf{U}^{(m)}_{q:D_q})]$, $\forall q$, $\forall m$, and $\mathbf{U}^{(m)}_q \in \mathbb{R}^{I_m \times R}_+$. In this way, the columns of matrix $\mathbf{U}^{(m)}_q$ are segmented into two parts: $\mathbf{U}^{(m)}_{q:C}$ represents the common subspace, while $\mathbf{U}^{(m)}_{q:D_q}$ represents the discriminative components to each tensor. The above common and discriminative

subspace discovery is the solution to the minimization of the following objective function:

$$J_0 = \sum_{q \in \{A,B\}} \frac{1}{n_q} \left\| \boldsymbol{\mathcal{X}}_q - [\![(\mathbf{U}^{(m)}_{q:C}, \mathbf{U}^{(m)}_{q:D_q})]\!] \right\|^2_F \tag{3.1}$$

where $\mathbf{U}^{(m)}_{q:C}$ and $\mathbf{U}^{(m)}_{q:D}$ are defined as above, n_q is the Frobenius norm of each tensor, and $\|\cdot\|^2_F$ stands for the Frobenious norm.

3.4.2 Regularized Shared and Discriminative Subspace Approach

Shared and discriminative subspace learning have also been explored in the context of nonnegative matrix factorization. In fact, CDNTF can be thought of as the extension of nonnegative shared subspace learning (JSNMF [74]) to higher dimensions. Under this framework, Gupta et al. [75] propose regularized nonnegative shared subspace learning that further imposes a mutual orthogonality constraint on the constituent subspace, which segregates the patterns. In the context of discovering common and discriminative mobility patterns, we extend the framework to **R**egularized **J**oint **S**ubspace **N**onnegative **T**ensor **F**actorization (RJSNTF) and with a slight abuse of notation, we derive the following minimization problem:

$$J_1 = J_0 + \sum_{m \in \{L,V,T\}} J_{R1}(\mathbf{U}^{(m)}_{q:C}, \mathbf{U}^{(m)}_{q:D_q}, \mathbf{U}^{(m)}_{q':D_{q'}}), \tag{3.2}$$

where $\mathbf{U}^{(m)}_{q:C}$ and $\mathbf{U}^{(m)}_{q:D_q}$ are defined as above. Focusing on our application, q' in Eq. 3.2 represents the time after the event that is different from q (time prior to the event) with $\mathbf{U}^{(m)}_{q:C} = \mathbf{U}^{(m)}_{q':C}$, and $\mathbf{U}^{(m)}_{q:D_q} \neq \mathbf{U}^{(m)}_{q':D'_q}$. Therefore, $J_{R1}(\cdot)$ is a regularization function used to penalize the "similarity" between subspaces spanned in $\{\mathbf{U}^{(m)}_q\}$ and $\{\mathbf{U}^{(m)}_{q'}\}$. Following [75], the mutually orthogonal constraints are defined as:

$$J_{R1} = \hat{\alpha} \left\| {\mathbf{U}^{(m)}_{q:C}}^T \mathbf{U}^{(m)}_{q:D_q} \right\|^2 + \hat{\beta} \left\| {\mathbf{U}^{(m)}_{q:C}}^T \mathbf{U}^{(m)}_{q':D_{q'}} \right\|^2 + \hat{\gamma} \left\| {\mathbf{U}^{(m)}_{q:D_q}}^T \mathbf{U}^{(m)}_{q':D_{q'}} \right\|^2, \tag{3.3}$$

where $\hat{\alpha}$, $\hat{\beta}$ and $\hat{\gamma}$ are the regularization parameters. When $J_{R1} = 0$, the model becomes identical to CDNTF.

RJSNTF enforces the shared components to be strictly identical, which is a hard constraint and might result in distortion during the factorization. Kim et al. [101] have proposed

the simultaneous discovery of common and discriminative topics via joint non-negative matrix factorization where this constraint is relaxed by redefining the regularization term. Their approach further emphasizes the similarities and differences of the common and discriminative components. Following the same idea and replacing $\mathbf{U}^{(m)}_{q:C}$ with $\mathbf{U}^{(m)}_{q:C_q}$ and $\mathbf{U}^{(m)}_{q':C_{q'}}$ to represent the similar components of tensors $\boldsymbol{\mathcal{X}}_q$ and $\boldsymbol{\mathcal{X}}_{q'}$. we derive **S**imultaneous **D**iscovery of **C**ommon and **D**iscriminative **N**onnegative **T**ensor **F**actorization (SDCDNTF) as the following minimization function:

$$J_2 = J_0 + \sum_{m\in\{L,V,T\}} J_{R2}(\mathbf{U}^{(m)}_{q:C_q}, \mathbf{U}^{(m)}_{q:D_q}, \mathbf{U}^{(m)}_{q':C_{q'}}, \mathbf{U}^{(m)}_{q':D_{q'}}), \tag{3.4}$$

and

$$J_{R2} = \alpha \left\| \mathbf{U}^{(m)}_{q:C_q} - \mathbf{U}^{(m)}_{q':C_{q'}} \right\|^2 + \beta \left\| \mathbf{U}^{(m)\,T}_{q:D_q} \mathbf{U}^{(m)}_{q':D_{q'}}) \right\|_{1,1}, \tag{3.5}$$

where $\|\cdot\|_{1,1}$ denotes the absolute sum of all the matrix entries.

3.4.3 Automatic Discovery of Discriminative Components

3.4.3.1 Our *PairFac* Formulation The above approaches fall under the same framework that splits the tensors' components into common and discriminative parts in advance, discovering these components with different regularization. These approaches require the number of shared (or distinct) components to be determined beforehand, which is difficult in practice. In this chapter, we propose a novel factorization method, which we term *PairFac*, that does not require manual input of the shared or distinct components. In order to achieve that, we assign a weight to each component that reflects the *discriminative coefficient* or *score* of the corresponding component.

For this purpose, we introduce two auxiliary data tensors $\boldsymbol{\mathcal{Z}}_B$ and $\boldsymbol{\mathcal{Z}}_A$ that represent the aggregated unique patterns found in each tensor respectively. We first define the following function to compute these auxiliary tensors.

Definition 2. Given a data tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1\times I_2\times I_3}$, $G(\boldsymbol{\mathcal{X}})$ is a *clamping* function that outputs a tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{I_1\times I_2\times I_3}$ that is derived from the input tensor $\boldsymbol{\mathcal{X}}$ with entries restricted to a given value range such that $\boldsymbol{\mathcal{Z}} = G(\boldsymbol{\mathcal{X}})$, where $G(\boldsymbol{\mathcal{X}})$ is defined as:

$$G(\boldsymbol{\mathcal{X}}) = \begin{cases} \boldsymbol{\mathcal{X}}_{i_1 i_2 i_3}, \text{if } \boldsymbol{\mathcal{X}}_{i_1 i_2 i_3} > \epsilon, \\ 0, \text{otherwise}, \end{cases} \tag{3.6}$$

where ϵ is a constant that defines the minimum entry in the tensor $\boldsymbol{\mathcal{Z}}$. ϵ can be empirically chosen to control the sparsity of the auxiliary tensors (we use $\epsilon = 0$ in this work). Note that the clamping function $G(\cdot)$ can also work with vectors and matrices. Then we compute $\boldsymbol{\mathcal{Z}}_q$ that captures the unique variance in $\boldsymbol{\mathcal{X}}_q$ from $\boldsymbol{\mathcal{X}}_{q'}$ as:

$$\boldsymbol{\mathcal{Z}}_q = G\left(\boldsymbol{\mathcal{X}}_q - \boldsymbol{\mathcal{X}}_{q'}\right), q \in \{A, B\} \tag{3.7}$$

Definition 3. Given two data tensors $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, $G'(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}})$ is a binary *clamping* function defined as:

$$G'(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}) = \begin{cases} 1, \text{if } |\boldsymbol{\mathcal{X}}_{i_1 i_2 i_3} - \boldsymbol{\mathcal{Y}}_{i_1 i_2 i_3}| < \epsilon^{'}, \\ 0, \text{otherwise}, \end{cases} \tag{3.8}$$

where $\epsilon^{'}$ is a constant that defines the minimum entry in the tensor $\boldsymbol{\mathcal{Z}}$. $\epsilon^{'}$ can also be empirically chosen to control the sparsity of the auxiliary tensors (we use $\epsilon^{'} = 0$ as well). Now with function $G^{'}$, we derive another auxiliary tensor $\mathcal{S}$ defined as:

$$\boldsymbol{\mathcal{S}}_{\boldsymbol{q}} = G^{'}\left(\boldsymbol{\mathcal{X}}_q, \boldsymbol{\mathcal{X}}_{q'}\right), q \in \{A, B\} \tag{3.9}$$

We further introduce the weight vectors $\mathbf{w}_q \in \mathbb{R}_+^R$ to capture the discriminative coefficient of each component. Given $\mathcal{S}$, we want to enforce $(1 - \mathbf{w}_q)$ to represent the contribution of the corresponding components to the common parts of the two tensors. Note that in Equation 3.9 we use a binary clamping function to infer the $\mathcal{S}$ tensor. This function captures the common factors as the ones whose differences are no larger than $\epsilon^{'}$. The choice of $\epsilon^{'}$ allows for imposing the degree of sparsity in the $\mathcal{S}$ tensor, which the stochastic version of optimization benefits from (introduced in Section 4.4). Besides, the binary clamping function enforces the non-common part to be zero such that the weights $(1 - \mathbf{w}_q)$ gives a clearer interpretation directly related to the degree of how much the pattern contributes towards to the common tensor. Our intuition is that while factorizing the original tensors into its latent patterns,

we would like to find a discriminative score for each pattern that corresponds to its unique contribution in each tensor. At the same time, we want to find a score that represents the commonality of a component in the two tensors. With the notations presented, we formally derive the minimization objective of *PairFac* as:

$$J_3 = J_0^{'} + J_{R3} + J_{R4}, \tag{3.10}$$

where $J_0^{'}$ differs from J_0 in that it does not require the manual split of common and discriminative parts in the factor matrix, and J_{R3} is a function to factorize the auxiliary tensors, defined as:

$$J_{R3} = \alpha \sum_{q \in \{A,B\}} \|\boldsymbol{\mathcal{Z}}_q - [\![\mathbf{w}_q; [\mathbf{U}_q]]\!]\|^2 + \beta \sum_{q \in \{A,B\}} \|\boldsymbol{\mathcal{S}}_q - [\![(1-\mathbf{w}_q); [\mathbf{U}_q]]\!]\|^2, \tag{3.11}$$

where $\mathbf{w}_q$ is the *level of discriminativeness* associated with component $\mathbf{U}_q$. According to Eq. 3.11, the degree of which $\mathbf{U}_q$ contributes towards the reconstruction of $\mathcal{Z}_q$ is determined by $\mathbf{w}_q$, and $\mathcal{Z}_q$ captures the information of predominant "discriminative" part between the two (before- and after-) tensors. Similarly, (1-$\mathbf{w}_q$) can be thought of as the degree of commonality associated with $\mathbf{U}_q$ as the reconstruction of $\mathcal{S}_q$ depends on $\mathbf{U}_q$ but weighted by (1-$\mathbf{w}_q$). Unlike $\mathcal{Z}_q$, $\mathcal{S}_q$ captures the information of predominant "common" part shared in both (before- and after-) tensors.

J_{R4} in Eq. 3.10 is a function to align the components in the order such that similar components should share similar weights as the result of the factorization:

$$J_{R4} = \gamma \sum_{m \in \{L,V,T\}} \sum_{j}^{R} \left\| (1-\mathbf{w}_{q_j})\mathbf{U}_{qj}^{(m)} - (1-\mathbf{w}_{q'_j})\mathbf{U}_{q'j}^{(m)} \right\|^2, \tag{3.12}$$

where $\mathbf{U}_{qj}^{(m)}$ is the jth column of factor matrix $\mathbf{U}_{qj}^{(m)}$ and $\mathbf{w}_{q_j}$ is its associated score.

Note that Eq. 3.10 differs from the objective defined in our prior work [231]. In [231], the objective is given as:

$$J_3^* = J_0^{'} + J_{R3}^*, \tag{3.13}$$

where

$$J^*_{R3} = \alpha \sum_{q \in \{A,B\}} \|\boldsymbol{\mathcal{Z}}_q - [\![\mathbf{w}_q; [\![\mathbf{U}_q]\!]]\!]\|^2 . \tag{3.14}$$

Compared to J^*_{R3} in Eq. 3.13, J_{R3} in Eq. 3.11 has the addition of a second term, which uses the auxiliary tensor $\mathcal{S}$. Our prior work attempts to model the level of uniqueness of each component i captured by the weight w_i. With the addition of $(1 - w_i)$, we can interpret it as the level of contribution to the commonality between the two tensors. Moreover, the output of Eq. 3.13 in our prior work [231] splits the tensor factors into common and discriminative components but is not able to identify directly the pair-wise common components across tensors. Previously, we addressed this problem through post-hoc analysis on examining the pair-wise similarity of the components, which could be cumbersome. In this study, we expand on our prior work by introducing J_{R4} to automatically align the common components in order.

To solve Eq. 3.10 we use the block coordinate descent method. Consider the updating of $\mathbf{U}_q^{(m)}$ at iteration k. Using the fact that if $\boldsymbol{\mathcal{X}}_q = \mathbf{U}_q^{(m)} \circ \mathbf{U}_q^{(m')} \circ \mathbf{U}_q^{(m'')}$, then $\boldsymbol{X}_{q(m)} = \mathbf{U}_q^{(m)} (\mathbf{U}_q^{(m'')} \odot \mathbf{U}_q^{(m')})^T$, where $\boldsymbol{X}_{q(m)}$ is the unfolded matrix of $\boldsymbol{\mathcal{X}}_q$ m-th mode. The objective (J_3) can be then re-written as:

$$\begin{aligned} \text{minimize } \frac{1}{2} \sum_{q \in \{A,B\}} \Big(& \frac{1}{n_q} \left\| \boldsymbol{X}_{q(m)} - \mathbf{U}_q^{(m)} (\mathbf{U}_q^{(m'')} \odot \mathbf{U}_q^{(m')})^T \right\|^2 \\ & + \alpha \left\| \boldsymbol{Z}_{q(m)} - \mathbf{U}_q^{(m)} \Lambda_{\mathbf{w}_q} (\mathbf{U}_q^{(m'')} \odot \mathbf{U}_q^{(m')})^T \right\|^2 \\ & + \beta \left\| \boldsymbol{S}_{q(m)} - \mathbf{U}_q^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_q}) (\mathbf{U}_q^{(m'')} \odot \mathbf{U}_q^{(m')})^T \right\|^2 \Big) \\ & + \gamma \sum_{m \in \{L,V,T\}} \left\| \mathbf{U}_q^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_q}) - \mathbf{U}_{q'}^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_{q'}}) \right\|^2 , \end{aligned} \tag{3.15}$$

where $\odot$ denotes the Khatri-Rao product, $\mathbf{I} \in \mathbb{R}^{R \times R}$ is the identity matrix, $\Lambda_{\mathbf{w}_q}$ is a diagonal matrix with $\mathbf{w}_q$ as its diagonal entries, and m' and m'' are used to index factor matrices other than $\mathbf{U}_q^{(m)}$. The gradient with respect to $\mathbf{U}_q^{(m)}$ is given as:

$$
\begin{aligned}
\bigtriangledown_{\mathbf{U}_q^{(m)}} J_3 = \frac{1}{n_q} & \left(\mathbf{U}_q^{(m)} (\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})^T - \boldsymbol{X}_{q_{(m)}} \right) (\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')}) \\
& + \alpha \left(\mathbf{U}_q^{(m)} \Lambda_{\mathbf{w}_q} (\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})^T - \boldsymbol{Z}_{q_{(m)}} \right) (\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')}) \Lambda_{\mathbf{w}_q}^T \\
& + \beta \left(\mathbf{U}_q^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_q}) (\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})^T - \boldsymbol{S}_{q_{(m)}} \right) (\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')}) (\mathbf{I} - \Lambda_{\mathbf{w}_q})^T \\
& + \gamma \left(\mathbf{U}_q^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_q}) - \mathbf{U}_{q'}^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_{q'}}) \right) (\mathbf{I} - \Lambda_{\mathbf{w}_q}).
\end{aligned}
\tag{3.16}
$$

Let

$$
\mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1} = \mathbf{U}_{q,k-1}^{(m')} \odot \mathbf{U}_{q,k-1}^{(m'')}. \tag{3.17}
$$

We take

$$
\mathcal{L}_{\mathbf{U}_q^{(m)}}^{k-1} = \left\| \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1\ T} \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1} \right\|^2, \tag{3.18}
$$

and

$$
\omega^{k-1} = \frac{\alpha_{k-1} - 1}{\alpha_k}, \tag{3.19}
$$

with $\alpha_0 = 1$ and

$$
\alpha_k = \frac{1 + \sqrt{4\alpha_{k-1}^2 + 1}}{2}. \tag{3.20}
$$

Furthermore, let

$$
\hat{\mathbf{U}}_{q,k-1}^{(m)} = \mathbf{U}_{q,k-1}^{(m)} + \omega_n^{k-1} \left(\mathbf{U}_{q,k-1}^{(m)} - \mathbf{U}_{q,k-2}^{(m)} \right), \tag{3.21}
$$

and

$$
\begin{aligned}
\hat{\mathbf{G}}_{\mathbf{U}_q^{(m)}}^{k-1} = \frac{1}{n_q} & \left(\hat{\mathbf{U}}_{q,k-1}^{(m)} \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1\ T} - \boldsymbol{X}_{q_{(m)}} \right) \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1} \\
& + \alpha \left(\hat{\mathbf{U}}_{q,k-1}^{(m)} \Lambda_{\mathbf{w}_q} \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1\ T} - \boldsymbol{Z}_{q_{(m)}} \right) \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1} \Lambda_{\mathbf{w}_q}^T \\
& + \beta \left(\hat{\mathbf{U}}_{q,k-1}^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_q}) \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1\ T} - \boldsymbol{S}_{q_{(m)}} \right) (\mathbf{1} - \Lambda_{\mathbf{w}_q}) \mathbf{F}_{\mathbf{U}_q^{(m)}}^{k-1} \\
& + \gamma \left(\hat{\mathbf{U}}_{q,k-1}^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_q}) - \hat{\mathbf{U}}_{q',k-1}^{(m)} (\mathbf{I} - \Lambda_{\mathbf{w}_{q'}}) \right) (\mathbf{I} - \Lambda_{\mathbf{w}_q})
\end{aligned}
\tag{3.22}
$$

be the gradient. Then we can derive the update based on [238]:

$$\mathbf{U}_{q,k}^{(m)} = \underset{\mathbf{U}_q^{(m)} \geq 0}{\operatorname{argmin}} \langle \hat{\mathbf{G}}_{\mathbf{U}_q^{(m)}}^{k-1}, \mathbf{U}_{q,k}^{(m)} - \hat{\mathbf{U}}_{q,k-1}^{(m)} \rangle + \frac{\mathcal{L}_{\mathbf{U}_q^{(m)}}^{k-1}}{2} \left\| \mathbf{U}_{q,k}^{(m)} - \hat{\mathbf{U}}_{q,k-1}^{(m)} \right\|_F, \tag{3.23}$$

which can be written in the closed form as

$$\mathbf{U}_{q,k}^{(m)} = \max \left(0, \hat{\mathbf{U}}_{q,k-1}^{(m)} - \hat{\mathbf{G}}_{\mathbf{U}_q^{(m)}}^{k-1} / \mathcal{L}_{\mathbf{U}_q^{(m)}}^{k-1} \right). \tag{3.24}$$

Similarly, let

$$\hat{\mathbf{w}}_{q,k-1} = \mathbf{w}_{q,k-1} + \omega^{k-1} \left(\mathbf{w}_{q,k-1} - \mathbf{w}_{q,k-2} \right), \tag{3.25}$$

and

$$\begin{aligned} \hat{\mathbf{G}}_{\mathbf{w}_q}^{k-1} = & \left(\hat{\mathbf{w}}_q \mathbf{F}_{\mathbf{w}_q}^{k-1^T} - \boldsymbol{\mathcal{Z}}_q \right) \mathbf{F}_{\mathbf{w}_q}^{k-1} - \left((\mathbf{1} - \hat{\mathbf{w}}_q) \mathbf{F}_{\mathbf{w}_q}^{k-1^T} - \boldsymbol{\mathcal{S}}_q \right) \mathbf{F}_{\mathbf{w}_q}^{k-1} \\ & - \sum_m \left(\mathbf{U}_{q,k-1}^{(m)^T} ((\mathbf{1} - \hat{\mathbf{w}}_q) \mathbf{U}_{q,k-1}^{(m)} - (\mathbf{1} - \Lambda_{\mathbf{w}_{q'}}) \mathbf{U}_{q',k-1}^{(m)}) \right), \end{aligned} \tag{3.26}$$

where

$$\mathbf{F}_{\mathbf{w}_q}^{k-1} = \mathbf{U}_{q,k-1}^{(m)} \odot \mathbf{U}_{q,k-1}^{(m')} \odot \mathbf{U}_{q,k-1}^{(m'')}. \tag{3.27}$$

Let

$$\mathcal{L}_{\mathbf{w}_q}^{k-1} = \left\| \mathbf{F}_{\mathbf{w}_q}^{k-1^T} \mathbf{F}_{\mathbf{w}_q}^{k-1} \right\|^2, \tag{3.28}$$

we can write the closed form of the update for $\mathbf{w}_q$

$$\mathbf{w}_{q,k} = \max \left(0, \hat{\mathbf{w}}_{q,k-1} - \hat{\mathbf{G}}_{\mathbf{w}_q}^{k-1} / \mathcal{L}_{\mathbf{w}_q}^{k-1} \right). \tag{3.29}$$

Algorithm 1 summarizes the above updating rules for solving Eq. 3.10 [2].

Convergence analysis We provide the convergence analysis of Algorithm 1. The convergence of alternating proximal gradient method is analyzed in [18].

[2]Our codes are publicly available at https://github.com/picsolab/pairfac.

ALGORITHM 1: `PairFac` algorithm for discovering the shared and discriminative subspace from tensor pairs.

Input : original tensors $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$, and $\mathbf{R}$.

Output: $\{w_q\}$, $\{U_q^{(m)}\}$ for $q \in \{A, B\}$ and $m \in \{L, V, T\}$

Compute $\boldsymbol{\mathcal{Z}}_q$ and $\boldsymbol{\mathcal{S}}_q$ by Eq. 3.7 and Eq. 3.9, $\forall m$;

Randomly initialize $\mathbf{U}_{q,-1}^{(m)} = \mathbf{U}_{q,0}^{(m)}$ and set $\mathbf{w}_{q,-1} = \mathbf{w}_{q,0} = [\frac{\mathbf{1}}{\mathbf{R}}]$, $\forall q$ and $\forall m$;

Set $\alpha_0 = 1$ and $k = 0$;

while *not converged* **do**

  $k = k + 1$;

  Compute $\mathcal{L}_{\mathbf{w}_q}^{k-1}$, $\mathcal{L}_{\mathbf{U}_q^{(m)}}^{k-1}$, and set ω^{k-1}, $\forall q$ and $\forall m$, according to Eq. 3.18, 3.28, 3.19;

  Compute $\hat{\mathbf{U}}_{q,k}^{(m)}$ and $\hat{\mathbf{w}}_{q,k}$, $\forall q$ and $\forall m$, according to Eq. 3.21, and 3.25;

  Update $\mathbf{U}_{q,k}^{(m)}$ and $\mathbf{w}_{q,k}$, $\forall q$ and $\forall m$, according to Eq. 3.23, and 3.29;

end

LEMMA 1. (Sufficient decrease property [24]). Let $f : \mathbb{R}^m \to \mathbb{R}$ be a continuously differentiable function with gradient $\bigtriangledown f$ assumed L_f-Lipschitz continuous and let $\sigma : \mathbb{R}^m \to (-\infty, +\infty]$ be a proper and lower semicontinuous function with $\inf_{\mathbb{R}^m}\sigma > -\infty$. For any t ¿ L_f and $u \in dom\sigma$, define

$$u^+ = \arg\min_x \{\langle x - u, \bigtriangledown f(u)\rangle + \frac{t}{2}\|x - u\|^2 + \sigma\ (u)\}. \tag{3.30}$$

Then we have that

$$f(u) + \sigma(u) - (f(u^+) + \sigma(u^+)) \geq \frac{1}{2}(t - L_f)\|u^+ - u\|^2. \tag{3.31}$$

LEMMA 2. Let $\Psi(\rho)$ be the objective function J_3 ,where $\rho = (\mathbf{U}_{q,k}^{(m)}, \mathbf{w}_{q,k})_{k\in\mathbb{N}}$ and $(\mathcal{L}_{\mathbf{U}_q^{(m)}}^k, \mathcal{L}_{\mathbf{w}_q}^k)_{k\in\mathbb{N}}$ are generated by our *PairFac* algorithm, we have that

$$\Psi(\mathbf{U}_{q,k}^{(m)}, \mathbf{w}_{q,k}) - \Psi(\mathbf{U}_{q,k+1}^{(m)}, \mathbf{w}_{q,k}) \geq \frac{\mathcal{L}_{\mathbf{U}_q^{(m)}}^k}{2}\|\mathbf{U}_{q,k}^{(m)} - \mathbf{U}_{q,k+1}^{(m)}\|^2, \forall m, \forall q,$$

$$\Psi(\mathbf{U}_{q,k+1}^{(m)}, \mathbf{w}_{q,k}) - \Psi(\mathbf{U}_{q,k+1}^{(m)}, \mathbf{w}_{q,k+1}) \geq \frac{\mathcal{L}_{\mathbf{w}_q}^k}{2} \|\mathbf{w}_{q,k} - \mathbf{w}_{q,k+1}\|^2, \forall q$$

Proof. The above inequalities can be obtained by using Lemma 1.□

In the following we show that the value of $\Psi(\rho)$ monotonically decreases on the sequence $(\rho^k)_k \in \mathbb{N}$, which is generated by *PairFac*.

LEMMA 3. Let $\Psi(\rho)$ be the objective function defined in J_3, where $\rho = (\mathbf{U}_{q,k}^{(m)}, \mathbf{w}_{q,k})$ and there exists $L > 0$ such that $\mathcal{L}_{\mathbf{U}_q^{(m)}}^k \geq L$ and $\mathcal{L}_{\mathbf{w}_q}^k \geq L$, then (i) The sequence $\{\Psi(\rho)\}_{k\in\mathbb{N}}$ is nonincreasing and for any $k \in \mathbb{N}$, there is a scalar $\beta > 0$ such that

$$\Psi(\rho^k) - \Psi(\rho^{k+1}) \geq \beta \|\rho^k - \rho^{k+1}\|_F^2, \forall k \geq 0.$$

(ii) We have

$$\sum_{k=1}^{\infty} (\|\mathbf{U}_{q,k+1}^{(m)} - \mathbf{U}_{q,k}^{(m)}\|^2 + \|\mathbf{w}_{q,k+1} - \mathbf{w}_{q,k}\|^2) = \sum_{k=1}^{\infty} \|\rho^{k+1} - \rho^k\|^2 < \infty, \tag{3.32}$$

and therefore the sequence $\{\Psi(\rho)\}_{k\in\mathbb{N}}$ is bounded.

Proof. Adding the inequalities from Lemma 2, we have

$$\Psi(\mathbf{U}_{q,k}^{(m)}, \mathbf{w}_{q,k}) - \Psi(\mathbf{U}_{q,k+1}^{(m)}, \mathbf{w}_{q,k+1}) \geq \frac{\mathcal{L}_{\mathbf{U}_q^{(m)}}^k}{2} \|\mathbf{U}_{q,k}^{(m)} - \mathbf{U}_{q,k+1}^{(m)}\|^2 + \frac{\mathcal{L}_{\mathbf{w}_q}^k}{2} \|\mathbf{w}_{q,k} - \mathbf{w}_{q,k+1}\|^2. \tag{3.33}$$

In *PairFac*, the Lipschitz constants $\mathcal{L}_{\mathbf{U}_q^{(m)}}^k \geq L$, $\mathcal{L}_{\mathbf{w}_q}^k \geq L$. Therefore, we have

$$\frac{\mathcal{L}_{\mathbf{U}_q^{(m)}}^k}{2} \|\mathbf{U}_{q,k}^{(m)} - \mathbf{U}_{q,k+1}^{(m)}\|^2 + \frac{\mathcal{L}_{\mathbf{w}_q}^k}{2} \|\mathbf{w}_{q,k} - \mathbf{w}_{q,k+1}\|^2 \geq \frac{L}{2} (\|\mathbf{U}_{q,k}^{(m)} - \mathbf{U}_{q,k+1}^{(m)}\|^2 + \|\mathbf{w}_{q,k} - \mathbf{w}_{q,k+1}\|^2). \tag{3.34}$$

Combining inequality 3.33 and 3.34 yields the following

$$\Psi(\rho^k) - \Psi(\rho^{k+1}) \geq \frac{L}{2} \|\rho^k - \rho^{k+1}\|^2. \tag{3.35}$$

Hence with $\beta = \min\{L/2, 1/2\}$, we prove (i).

From Eq. 3.33 we obtain that the sequence $\{\Psi(\rho)\}_{k\in\mathbb{N}}$ is nonincreasing. Since Ψ is assumed to be bounded from below by *zero*, it converges to some real number $\bar{\Psi}$. Let N be a positive integer. Summing up all $k \geq 1$ for inequality 3.35, we have

$$\begin{aligned}\Psi(\rho^0) - \bar{\Psi} &\geq \Psi(\rho^0) - \Psi(\rho^N) \\ &\geq \frac{L}{2}\sum_{k=1}^{N} \|\rho^k - \rho^{k+1}\|^2 \\ &= \frac{L}{2}\sum_{k=1}^{N} (\|\mathbf{U}_{q,k}^{(m)} - \mathbf{U}_{q,k+1}^{(m)}\|^2 + \|\mathbf{w}_{q,k} - \mathbf{w}_{q,k+1}\|^2)\end{aligned} \tag{3.36}$$

Taking the limit as $N \to \infty$, we prove the assertion (ii).

Based on this lemma, we then provide a convergence result of algorithm 1 under certain assumptions. Let $\rho = (\mathbf{U}_q^{(m)}, \mathbf{w}_q)$. A point ρ satisfies the KKT-condition for the solution to Eq. 3.10 if

$$\begin{aligned}\frac{1}{n_q}\mathbf{U}_q^{(m)} \star \Bigg(&\left(\mathbf{U}_q^{(m)}(\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})^T - \boldsymbol{X}_{q_{(m)}}\right)(\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')}) \\ &+\alpha\left(\mathbf{U}_q^{(m)}\Lambda_{\mathbf{w}_q}(\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})^T - \boldsymbol{Z}_{q_{(m)}}\right)(\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})\Lambda_{\mathbf{w}_q}^T \\ &+\beta\left(\mathbf{U}_q^{(m)}(\mathbf{I} - \Lambda_{\mathbf{w}_q})(\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})^T - \boldsymbol{S}_{q_{(m)}}\right)(\mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')})(\mathbf{I} - \Lambda_{\mathbf{w}_q})^T \\ &+\gamma\left(\mathbf{U}_q^{(m)}(\mathbf{I} - \Lambda_{\mathbf{w}_q}) - \mathbf{U}_{q'}^{(m)}(\mathbf{I} - \Lambda_{\mathbf{w}_{q'}})\right)(\mathbf{I} - \Lambda_{\mathbf{w}_q})\Bigg) = 0, \\ \mathbf{w}_q \star \Bigg(&\left(\mathbf{w}_q{\mathbf{F}_{\mathbf{w}_q}}^T - \boldsymbol{\mathcal{Z}}_q\right)\mathbf{F}_{\mathbf{w}_q} - \left((\mathbf{1} - \mathbf{w}_q){\mathbf{F}_{\mathbf{w}_q}}^T - \boldsymbol{\mathcal{S}}_q\right)\mathbf{F}_{\mathbf{w}_q} \\ &-\sum_m \left({\mathbf{U}_q^{(m)}}^T((\mathbf{1} - \mathbf{w}_q)\mathbf{U}_q^{(m)} - (\mathbf{1} - \Lambda_{\mathbf{w}_{q'}})\mathbf{U}_{q'}^{(m)})\right)\Bigg) = 0,\end{aligned} \tag{3.37}$$

where $\star$ denotes component-wise product and $\mathbf{F}_{\mathbf{w}_q}$ is defined in Eq. 3.27.

THEOREM 1. Suppose the sequence $\{\rho = (\mathbf{U}_{q,k}^{(m)}, \mathbf{w}_{q,k})\}$ generated by algorithm *PairFac* is uniformly away from zero, i.e., there exists $\mathcal{L} > 0$ such that $\mathcal{L}^k_{\mathbf{U}_q^{(m)}} \geq L$ and $\mathcal{L}^k_{\mathbf{w}_q} \geq L$, $\forall q$ and $\forall m$. Then any limit point of $\{\rho\}$ satisfies the KKT-conditions 3.37.

The proof of Theorem 1 is provided in Appendix.

3.4.4 Parallel Implementation

In this section, we provide a scalable implementation for the *PairFac* algorithm. Our method is based on FlexiFaCT [20], which is a MapReduce algorithm for PARAFAC and coupled PARAFAC decompositions.

The key idea of FlexiFaCT is to split the tensor data into multiple blocks, each of which is further split into smaller blocks with no shared rows or columns. Given the complex nature of tensorial computation, researchers have initiated efforts in devising more efficient algorithms for tensor computations, e.g., GigaTensor [96], FlexiFaCT [20], MET [106], Turbo-SMT [169], and Haten2 [94]. We adopt the scheme introduced in [20] due to its simplicity in implementation as well as its ability for coupled tensor matrix factorization. The parallelization implementation involves three steps:

Step 1: Blocking for Parallelization. This step is to partition the data tensors into certain blocks so that each block could run in parallel. Following [20], we term one set of independent blocks in the corresponding tensor a *stratum*, and then we denote the number of blocks in each stratum by d. To have full coverage of the whole tensor, we require d^2 strata. For a stratum s we have blocks $P_i^{(s)}$ for $i = 0...d-1$. Let each block P be the tensor that contains all data observations in (b_i, b_j, b_k) where b_i, b_j, b_k are ranges in I, J, and K: $b_i = (i\lceil I/d\rceil, (i+1)\lceil I/d\rceil)$, $b_j = (j\lceil I/d\rceil, (j+1)\lceil I/d\rceil)$, $b_k = (k\lceil I/d\rceil, (k+1)\lceil I/d\rceil)$. With this we define the blocks for stratum s as

$$
\begin{aligned}
P_i^{(s)} &= (b_i, b_{j_{s,i}}, b_{k_{s,i}}) \\
j_{s,i} &= (i+s) \bmod d \\
k_{s,i} &= \lfloor (i+s/d) \rfloor \bmod d,
\end{aligned}
$$

for $i = 0...d-1$.

Step 2: Parallelizing the Computation. We partition the original tensors as well as the three auxiliary tensors with the same schema so that $P_i^{(s)}$ denotes the same block across different tensors. With this partition schema, we run the strata sequentially, but for each stratum we compute the gradient with respect to $\mathbf{U}_q^{(m)}$ by Eq. 3.22 and to $\mathbf{w}_q$ by Eq. 3.26 based on sparse tensors constructed from (b_i, b_j, b_k) in parallel on d machines.

Step 3: Gradient Summation. Now we have temporary gradient values computed by each machine. These values are sent the partial gradients to the centralized master server. Lastly, the final gradients in Eq. 3.22 and Eq. 3.26 are the summation of all the partial gradients.

In practice, step 1 can be regarded as a prepossessing step to index the observations in the tensors to certain blocks for parallelization. We can run step 2 and 3 repeatedly, iteratively updating $\mathbf{U}_q^{(m)}$ and $\mathbf{w}_q$, $\forall m$ and $\forall q$, until the algorithm converges.

3.5 EVALUATION

In this section, we provide the evaluation of *PairFac* based on a synthetic dataset. Section 5.1 describes the synthetic dataset, while Section 5.2 illustrates the exemplary output of *PairFac*. Section 5.3 provides the quantitative comparison with existing baselines. Since *PairFac* outputs components with a list of associated weights instead, Section 5.4 discusses several approaches to identify the common and discriminative components based on the weights. Finally, in Section 5.5, we provide guidance on the parallelized implementation of *PairFac*.

3.5.1 Synthetic Data Setup

The synthetic dataset generation aims to provide multidimensional datasets that share some signals in common. To this end, we want to generate two three-way tensors $\boldsymbol{\mathcal{X}}_B \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ and $\boldsymbol{\mathcal{X}}_A \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ according to the equation $\boldsymbol{\mathcal{X}}_B = \sum_{r=1}^{R} \mathbf{U}_{B,r}^{(L)} \circ \mathbf{U}_{B,r}^{(T)} \circ \mathbf{U}_{B,r}^{(V)}$ and $\boldsymbol{\mathcal{X}}_A = \sum_{r=1}^{R} \mathbf{U}_{A,r}^{(L)} \circ \mathbf{U}_{A,r}^{(T)} \circ \mathbf{U}_{A,r}^{(V)}$, where $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$ share the first K components among the total R components in the first factor matrix and have exactly the same columns in the second and third factor matrices. K is a parameter that controls the extent to which the two generated tensors are similar to each other. Our generation rules of the synthetic dataset follow the idea in [101]. The shared part in the first factor matrix are generated as:

$$\mathbf{U}_{C,r}^{(L)} = \begin{cases} 1, sr \leqslant m < s(r+1), \\ 0, \text{otherwise}, \end{cases}$$

where $s = I/\ (R + (R - K))$, r is the column index for each matrix and m is the row index. We generate the discriminative parts in the first factor matrix as:

$$\mathbf{U}^{(L)}_{D:B,r} = \begin{cases} 1, sK + sr \leqslant m < sK + s(r+1), \\ 0, \text{otherwise}, \end{cases}$$

and

$$\mathbf{U}^{(L)}_{D:A,r} = \begin{cases} 1, sR + sr \leqslant m < sR + s(r+1), \\ 0, \text{otherwise}. \end{cases}$$

In addition, each row of $\{\mathbf{U}^{(T)}\}$ and $\{\mathbf{U}^{(V)}\}$ is set to be a unit vector with only one non-zero entry at a randomly selected dimension. We further add sparse Gaussian noise $\mathcal{N}(0, \sigma^2)$ with different levels of variance to 20% of the entries in $\mathbf{U}^{(L)}_B$ and $\mathbf{U}^{(L)}_A$.

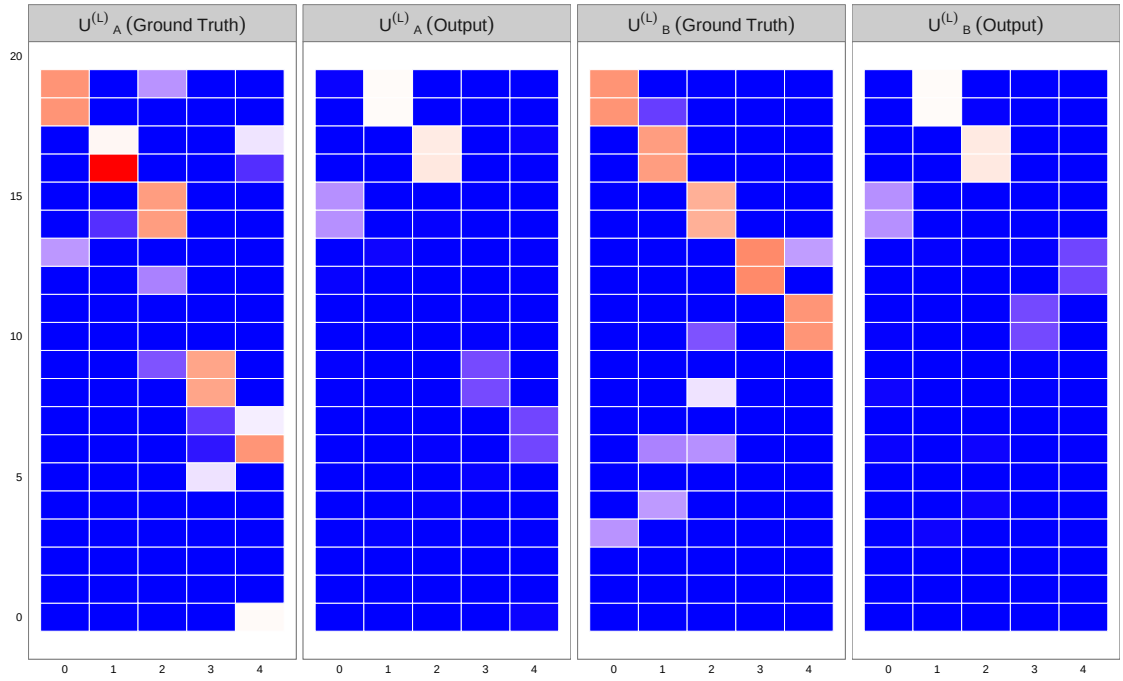

Figure 4: Illustration of *PairFac* Output. We reorder the components of each output factor matrix by its associated weight in ascending order from left to right. The weight vector $\mathbf{w}_A = [.0001, .0000, .0000, .4874, .5124]$ and the weight vector $\mathbf{w}_B = [.0000, .0003, .0013, .4877, .5107]$.

3.5.2 Algorithm Output Illustration

We first provide the illustration of the output from our approach with the synthetic dataset generated by setting $I_1 = I_2 = I_3 = 20$, $R = 5$, $K = 3$, $\sigma^2 = 0.1$ and $\alpha = 10^{-5}$, $\beta = 2$, and $\gamma = 10^{-4}$. Fig. 4 shows an illustrative example of the factor matrices obtained from our method in comparison with the ground truth factor matrices. Each column of the output factor matrices is associated with a discriminative score (i.e., $\mathbf{w}_q$ as in Eq. 3.11). To reiterate, a higher score represents a greater level of discriminativeness for the corresponding component in comparison with the components in the factor matrix in the second tensor. As explained in Eq. 3.11, the value of each $\mathbf{w}_q$ reflects the extent to which its corresponding component contributes to the reconstruction of the "differing" part between the two (before and after) tensors. We observe that our method nicely segments each output factor into two parts based on the learned weights. The weights of the common components are almost zero while the discriminative components contribute equally to the overall discriminative power. One outstanding property of this model compared to our prior work [231] is its ability to align the similar components in the corresponding order. For instance, we observe that the first three columns are common components in $\mathbf{U}_A^{(L)}$(Output) and $\mathbf{U}_B^{(L)}$(Output). Among these three columns, the first columns in these two matrices correspond to one common component (the third column) in $\mathbf{U}_A^{(L)}$(Ground Truth) and $\mathbf{U}_B^{(L)}$(Ground Truth). Similarly, we could find that the second and the third columns in the output matrices concur with themselves and can also find their matches in the ground truth matrices.

3.5.3 Comparisons with Baselines

As discussed in Section 4, there are three existing models that we adopt for comparisons, including CDNTF [128], our extension of RSJNMF [75] to RSJNTF, and our extension of SDCDNMF [101] to SDCDNTF.

3.5.3.1 Baselines We include three baselines and one modification of our method for comparative studies:

- CDNTF [128] takes an input K and splits the factor matrix into K common components

and $(R-K)$ discriminative components by solving Eq. 3.1 with multiplicative updating rules.

- RSJNTF is our tensor extension of RSJNMF [75]. It also requires the number of common components K as input K and is based on a similar framework with CDNTF where additional mutually orthogonal constraints on the common and discriminative components are added. We develop multiplicative updating rules to solve Eq. 3.2.
- SDCDNTF is our tensor extension of SDCDNMF [101], which also requires K as input. It can be classified under the same framework as RSJNTF, where there is a relaxation to the constraints on the shared components. We extend the block coordinate descent framework to SDCDNTF to solve Eq. 3.4.
- *PairFac* does not require the specification of K. Instead, it generates two weight vectors that represent the discriminative scores for each of its components.

3.5.3.2 Evaluation Metrics To quantitatively evaluate the performance of our proposed approach in comparison with existing literature, we use three measures, namely, (a) the relative reconstruction error, (b) the quality of the recovered discriminative components and (c) the quality of the recovered common components. To measure the quality of the reconstruction, we compute the relative reconstruction error as:

$$\frac{1}{2}\Big(\frac{\left\|\boldsymbol{\mathcal{X}}_B - [\![\mathbf{U}_B^{(L)}, \mathbf{U}_B^{(T)}, \mathbf{U}_B^{(V)}]\!]\right\|^2}{\|\boldsymbol{\mathcal{X}}_B\|^2} + \frac{\left\|\boldsymbol{\mathcal{X}}_A - [\![\mathbf{U}_A^{(L)}, \mathbf{U}_A^{(T)}, \mathbf{U}_A^{(V)}]\!]\right\|^2}{\|\boldsymbol{\mathcal{X}}_A\|^2}\Big).$$

The quality of the recovered discriminative part of the factor matrix is computed as the similarity between the output factor matrix and the ground truth factor matrix: $sim_D(\mathbf{U}, \bar{\mathbf{U}}) = \frac{1}{R-K}\sum_{r>K}^{R}\cos(\mathbf{U}_r, \bar{\mathbf{U}}_r) = \frac{\mathbf{U}_r \cdot \bar{\mathbf{U}}_r}{\|\mathbf{U}_r\|\|\bar{\mathbf{U}}_r\|}$, where $\mathbf{U}_r$ is the r-th discriminative component in the ground truth factor matrix and $\bar{\mathbf{U}}_r$ is the output of the r-th discriminative component. Because there is an ambiguity in the column ordering [3], we try out all possible permutations of $R-\kappa$ components and compute the maximum similarity. Furthermore, we compute the maximum similarity score of the common components as: $sim_C(\mathbf{U}, \bar{\mathbf{U}}) = \frac{1}{R}\sum_{r\leqslant K}^{R}\cos(\mathbf{U}_r, \bar{\mathbf{U}}_r) = \frac{\mathbf{U}_r \cdot \bar{\mathbf{U}}_r}{\|\mathbf{U}_r\|\|\bar{\mathbf{U}}_r\|}$.

3.5.3.3 Experiment Setup Following the setup introduced in section 5.1, we generate another synthetic dataset by setting $I_1 = 100$, $I_2 = 10$, $I_3 = 20$, $\sigma^2 = 0.5$, $R = 10$, and $K = 5$.

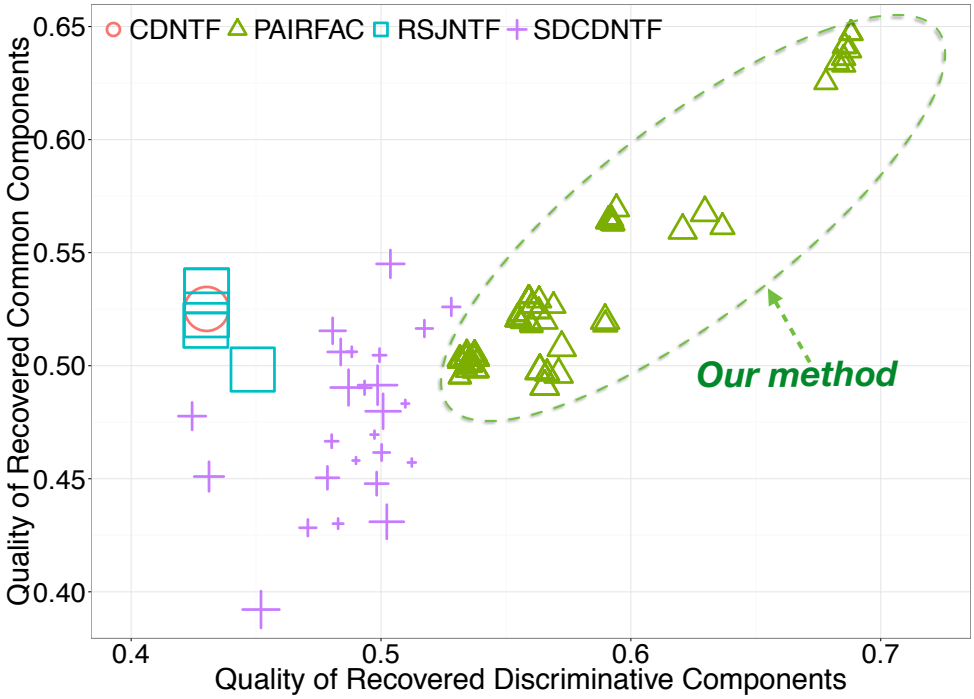

Figure 5: Comparison of *PairFac* with existing methods. Each point represents the average score of 30 runs for each combination of the parameter setting. The size of points represents the reconstruction error.

For SDCDNTF, we experiment with α and $\beta \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. For RSJNTF, following [75], we set a super parameter α in the same range. Finally, for *PairFac*, we set α and $\beta \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$, and $\gamma \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$. We plot the average reconstruction error versus the average similarity score on the discriminative components as well as on the common components from 30 runs of each method on every set of parameters.

3.5.3.4 Results Fig. 5 presents the comparison of the various methods from 30 independent trials for each combination of parameter settings. The x-axis and y-axis show the quality of recovered discriminative components and the quality of recovered common components.

Each point represents the average result of 30 runs. The size of each point is proportional to the reconstruction error. We observe that *PairFac* has comparable reconstruction quality with that of SDCDNTF. We also notice that most of the points from *PairFac* lay on the top-right region in the figure, exhibiting higher quality in both recovered discriminative and common components. We conducted additional experiments on cases where the data have a varying number of modes that are similar or different. Our results show that *PairFac* consistently achieves better recovery quality in both the common and discriminative components. The results are included in the appendix.

3.5.4 Identification of Common and Discriminative Patterns

PairFac learns the ranked components based on their discriminative scores. Components that have higher similarities associate with low weights. In this section, we show how to identify common and discriminative patterns.

Given a vector of ranked numerical values in the range of $(0, 1)$ generated by *PairFac*, the problem of identifying common and discriminative components is equivalent to searching for a proper threshold θ, such that components with $w < \theta$ would be regarded as common components, while the rest can be regarded as discriminative components. We experimented with four approaches for the selection of a cutoff threshold:

- *Fixed threshold.* The simplest approach is to define a fixed threshold, regardless how many common components are in the tensors. We can set $\theta = \frac{1}{R}$, which essentially makes the assumption that every component (from the R components in total) has equal probability of being discriminative.
- *Largest Difference.* We could also define θ as the maximum difference between two consecutive (ordered) weights.
- *Two Clusters.* The weights learned from *PairFac* tend to fall into two natural groups. Therefore small weights and large weights are likely to be separated by a simple one-dimensional clustering with two clusters.
- *Bimodal Density.* Given that the weights tend to fall into two natural groups, we could model the distribution of weights using a kernel density function and set θ equal to the

local minimum of the area between two peaks.

3.5.4.1 Experimental Setup In this experiment, we aim into evaluating the number of common components identified by different heuristics. Following the setup introduced in section 5.1, we generated another synthetic dataset by setting $I_1 = 100$, $I_2 = 10$, $I_3 = 20$, $\sigma^2 = 0.5$, $R = 10$, and $K \in \{1, 2, 4, 5, 6, 7, 8, 9\}$. We perform five runs with each value of K and reported the run with the best results.

3.5.4.2 Results In Fig. 6, we present the number of common components identified based on the value θ defined by the different heuristics aforementioned. Ideally, for a *perfect* choice of θ, we expect the results to lay on the line $y = x$. Of the four approaches attempted, we observe that the value of θ defined by *bimodal density* and *largest differences* are the closest to the *optimal* solution.

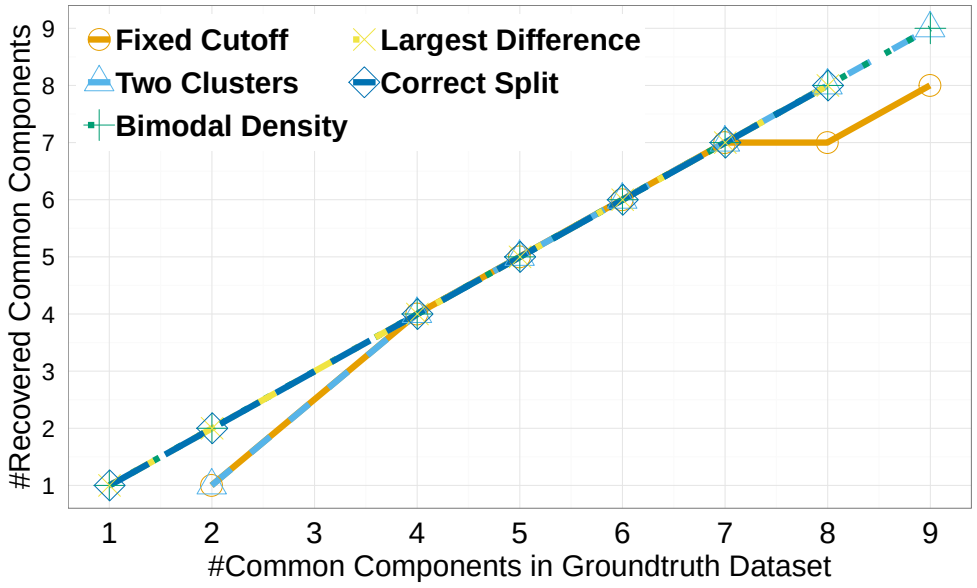

Figure 6: Number of common components identified by different heuristic approaches. Dark blue line with diamond-shaped points denotes the perfect split between the common and discriminative components; the cutoff defined by Bimodal Density (green line with cross-shaped points) has the closest split with the optimal split.

3.5.5 Parameter Sensitivity

In our approach, parameters α and β control the weight placed on identifying the discriminative or common components, and γ controls the extent to which common components could be aligned together. In this section, we evaluate the sensitivity of our approach with regards to these parameters.

3.5.5.1 Experimental Setup We follow the same experimental setup as introduced in Section 5.2 for *PairFac*. For each experiment, we vary one of the parameters α, β, and γ in *PairFac*, while keeping the remaining parameters constant.

3.5.5.2 Evaluation Metrics In Eq. 3.10, we introduced auxiliary tensors to capture the common as well as the unique parts of both tensors. α and β control the importance of the discriminative and common components respectively. As *PairFac* learns the discriminative weights of each component we label them in order to classify them as common or unique. During this process, we need to identify a cutoff point for the (ranked) weights. The components that have discriminative power higher than this cutoff would be regarded as unique patterns to each tensor. Section 5.4 suggests that the distribution of weights follow a bi-modal distribution and the local minimum of the pit is the optimal cutoff for the split. Hence, we separate the components using a bimodal distribution for the weights. To measure the extent to which the bimodal distribution could reach a clear separation, we compute the bimodal separation index [249]. Furthermore, the third term in Eq. 3.10 enforces that similar components should be aligned together. γ is expected to control the degree to which the factorization should be constrained by the component similarity regularization.

3.5.5.3 Results For evaluating the sensitivity of α and β, we calculate their impact on the separability and the relative reconstruction error, with a fixed value of γ. For evaluating the sensitivity of γ, we calculate its effect on the similarity of the common components and the relative reconstruction error, with α and β fixed. We run *PairFac* with each parameter setting for 30 runs and report the average measures with standard errors.

Effect of auxiliary tensors. We vary the weight of factorizing the auxiliary tensors by setting α and $\beta \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$, and $\gamma = 1$. Fig. 7 shows the average relative reconstruction given different settings of α and β. The results suggest that with the increase of the weight for factorizing the auxiliary tensors, the reconstruction quality degrades. One exception is shown in Fig. 7 (a), where the relative reconstruction error decreases while α becomes larger. However, we expect the factorization quality would eventually go up with larger α values. Fig. 8 shows the average separability with different parameter settings of α and β when γ is fixed to 1. When β is fixed, the separability becomes larger when α increases, except when β is equal to 1. When α is fixed, we observe that the separability decreases first and then increases.

Effect of column regularization. We vary the weight of enforcing the column similarity regularization by setting $\gamma \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$ and $\alpha = \beta = 10^{-6}$. Fig. 9 (a) shows the average relative reconstruction error with different values of γ. As we observe, when $\gamma \leqslant 10^{-2}$, the column regularization barely draws any impacts on the reconstruction error, although we have gains in the similarity between the resultant common components as shown in Fig. 9 (b). When $\gamma \geq 10^{-2}$, the reconstruction error first decreases and increases again, while the similarity scores seem to continue rising with the increase of γ. It is possible that a reasonably large choice of γ can give rise to the importance of column regularization in the factorization steps. However, when γ is set to be too large, the factorization result would bias towards making excessive agreements between the common components, while losing its quality on the true discriminative patterns.

To summarize, we demonstrated that, in practice, the "relative reconstruction error" can be used to observe the appropriate range of the parameter settings. For example, in our experiments, we found that the reconstruction errors are relatively stable for a wide range of α and β values, except for a very large value in either of the two parameters (Fig. 7 and 8). γ controls the level of "similarity" in common components, which is a parameter that allows the algorithm to adapt to different application scenarios (Fig. 9(b)). A too large value of γ (too much tolerance of "similarity") may degrade the reconstruction results, which can be easily discovered from plotting the reconstruction error against γ (Fig. 9(a)).

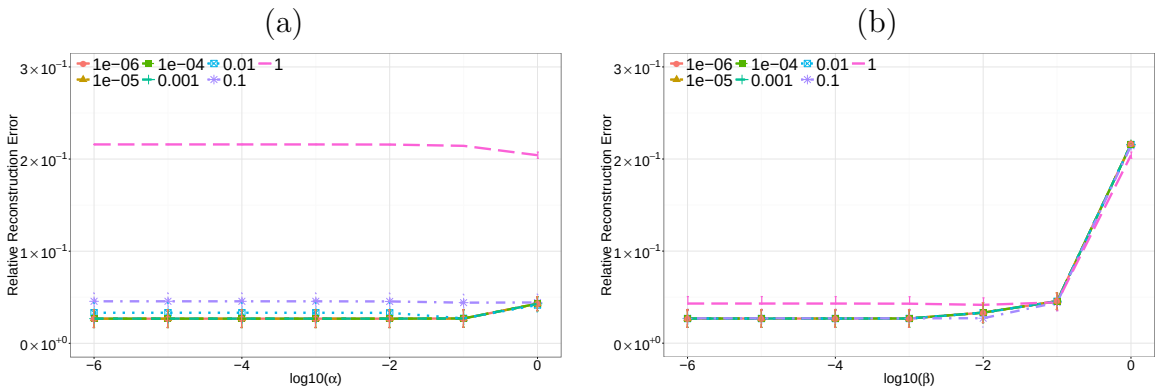

Figure 7: Parameter Sensitivity Analysis of *PairFac* (1). (a) α vs. relative construction error (b) β vs. relative reconstruction error. Different lines represent the settings of different α or β values. (a) shows that as α goes large, we have higher reconstruction errors except when $\beta = 1$; (b) shows that as β larger tend to lead to higher reconstruction errors.

3.5.6 Scalability

In this section, we provide the scalability analysis of our proposed method in terms of parallel and non-parallel implementations. The purpose of the experiments on the synthetic data is to demonstrate the run-time efficiency of the proposed method as well as the speedup of the parallelization strategy. To understand how different tensor properties affect the computation time, we perform a set of experiments with varying conditions. There are three sets of parameters involved in this analysis: observations N is the number of nonzero elements in the tensor; dimensionality I is the size of a mode; and rank R is the minimal number of rank one tensors, which generate the tensor as their sum.

3.5.6.1 Experiments We construct two synthetic tensors following the dataset setup introduced in Section 5.1, with a varying set of parameters to test the scalability with respect to each of them. To this end, we fix two of three parameters N, I, R and vary the remaining one. We conducted three experiments for the sake of validating the scalability of our method concerning the number of observations, the dimensionality of the tensors, and

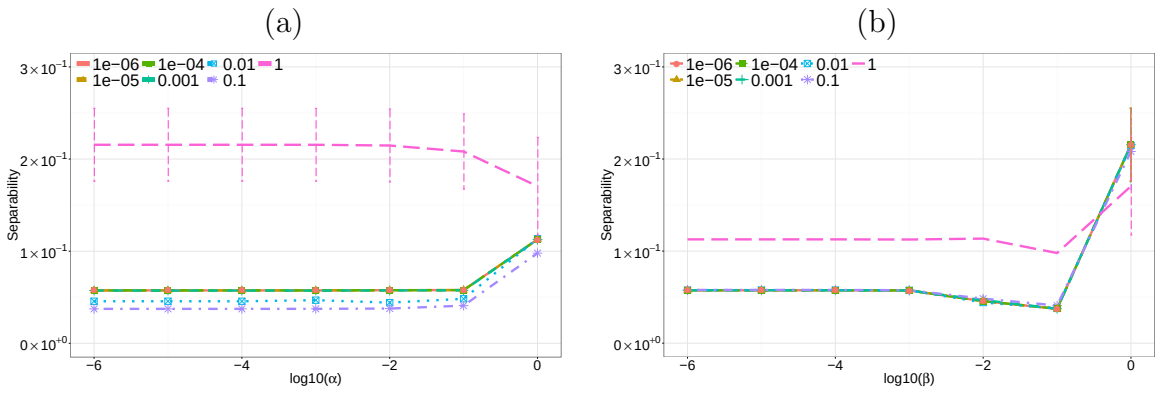

Figure 8: Parameter Sensitivity Analysis of *PairFac* (2). (a) α vs. separability (b) β vs. separability. Different lines represent the settings of different α or β values. (a) shows that as α goes larger, the separability becomes larger except when $\beta = 1$; (b) shows that as β becomes larger, however followed by in increasing trend.

the tensor rank. Unless otherwise stated, we set the convergence criteria as either reaching 10,000 iterations or the relative reconstruction is below 10^{-4}.

Observations. We generate a synthetic dataset with $I_1 = I_2 = I_3 = 1000$, $R = 30$, $K = 10$, following section 5.1. Then we take the top N largest elements from each tensor to construct the sparse tensor, where N varies in the range of $\{10^2, 10^3, 10^4, 10^5, 10^6\}$. We set $R = 10$ as the number of components after the factorization. In Fig. 10 (a), we show the running time of our algorithm against the number of observations.

Dimensionality. We generate a synthetic dataset with $I_1 = I_2 = I_3 \in \{400, 500, 600, 700, 800\}$, $R = 30$, $K = 10$. Then we take the top 10^4 largest elements from each tensor to construct the sparse tensors and set $R = 10$ as the number of components after the factorization. In Fig. 10 (b), we show the running time of our algorithm against the dimensionality.

Rank. We generate a synthetic dataset with $I_1 = I_2 = I_3 = 1000$, $R = 30$, $K = 10$. Then we take the top 10^5 largest elements from each tensor to construct the sparse tensors and set R in the range of $\{10, 20, 30, 40, 50\}$ as the number of components after the factorization. In Fig. 10 (c), we show the running time of our algorithm against the rank of the tensor

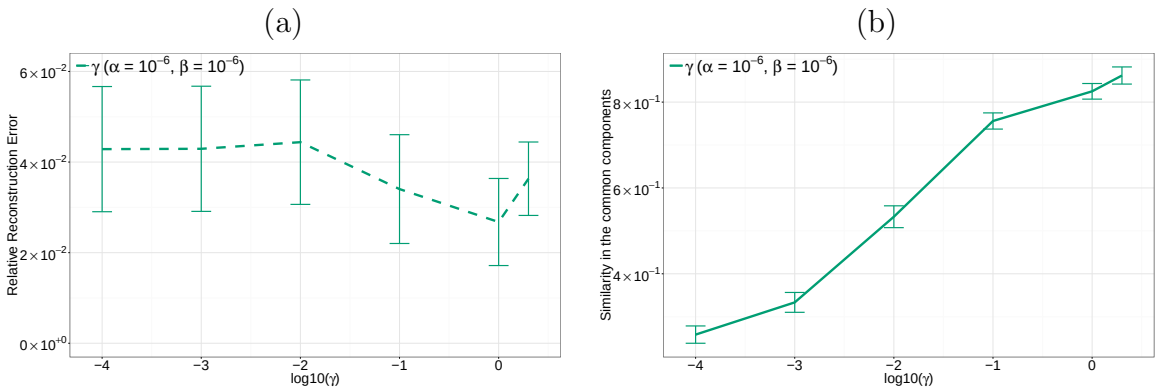

Figure 9: Parameter Sensitivity Analysis of *PairFac* (3). (a) γ vs. relative reconstruction error (b) γ vs. similarity in the common components. (a) shows that as γ goes large, the relative reconstruction error decreases and then goes up after taking certain larger values; (b) shows that as γ increases, we have higher similarities in the common components.

decomposition.

3.5.6.2 Results The results show that the running time of *PairFac* scales reasonably well with the growth of the number of observations, the dimensionality of the tensors, and the number of components. Furthermore, with the stratum split mechanism introduced in Section 4.4, we could reach better scalability with the help of multi-threading processing of *PairFac*. The yellow lines in Fig 10 show the running time with two threads in comparison to single-threaded *PairFac*.

3.6 CASE STUDIES

In this section, we illustrate the application of our method in two case studies, which showcases the effects of specific events in the urban space, including the Paris terrorist attacks and the Thanksgiving holiday weeks comparison in New York City (NYC).

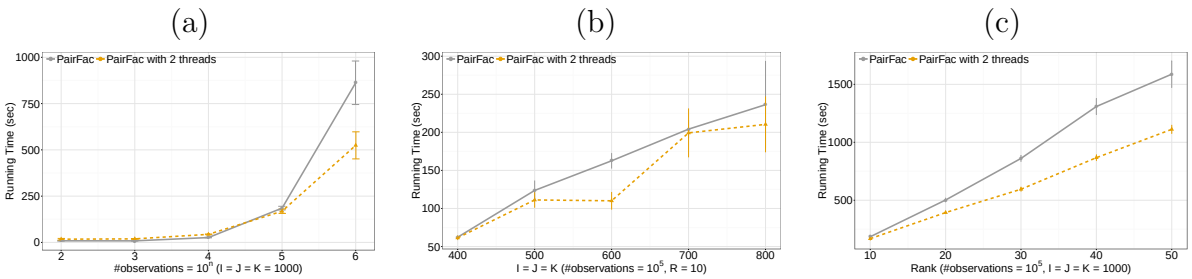

Figure 10: Scalability Analysis of *PairFac*. (a) number of observations vs. Running Time (b) Dimensionality vs. Running Time and (c) Rank vs. Running Time.

Table 4: Data Sources Used in the Case Study of Paris Terrorist Attacks.

Type of Data	Dimensions extracted	Volume of raw data extracted
Traffic Sensor databases	Location, Time	10,915,272 hourly occupancy rate from 2,885 road sensors
Check-ins and POI database	Location, Time, Activity	86,033 check-ins with 15,375 POI information
Geo-tagged Tweets	Location, Time	121,631 tweets

3.6.1 Paris Attacks

In this section, we use *PairFac* to analyze the effects of the Paris terrorist attacks in the surrounding urban space. In our previous study, we investigated the immediate impact of urban mobility in the following week of the attacks. In this study, we collect Twitter check-ins and traffic sensor data in the month following the attacks from the Paris area and apply our approach to study the long-term impacts on urban mobility.

3.6.1.1 Dataset Table 4 summarizes the data sources we used for our case study. The first dataset is the geo-tagged tweets from Paris collected through the Twitter API be-

tween the period of Oct 16th, 2015 and Dec 18, 2015. The region is defined by a rectangle boundary [3] that covers the Paris area. 121,631 geo-located tweets were extracted during the period covered. The second dataset includes approximately 10.9 million records of traffic sensor data [45]. It provides the hourly occupancy rate of 2,889 road segments in the area of Paris and covers the same period as above. Our third dataset is from Foursquare collected by Yang et al. [241] and it contains 86,033 check-ins from 15,375 POIs in the area of Paris between April 2012 and September 2013.

3.6.1.2 Case Study Setup In our previous study, we used grid-cell based city partition to study the immediate impact of the terrorist attacks. We constructed three-mode tensors, where the three dimensions are location, time, and venue type, respectively. While the spatial locations can be represented via a two-dimensional variable, e.g., (x, y) or (latitude, longitude), they can also be represented as a list of locations indexed by the two-dimensional variable. We use the latter representation in our experiment to facilitate the interpretation of discovered impact in terms of "location mode" and to compare it with other modes. In our case studies, we used the neighborhoods to construct a list of locations as one mode in the input tensors, where each entry in the location dimension represents one neighborhood location. We extract 80 quartiers from 20 arrondissements in Paris as the possible values of the location dimension. For the temporal dimension, we segment a week into $24 \times 7 = 168$ hourly intervals. Finally, for the venue dimension, we extract the nine primary categories in the Foursquare venue hierarchy that includes *Professional & Other Places* (POP), *Travel & Transport* (TT), *Food* (F), *Outdoors & Recreation* (OR), *Nightlife Spot* (NS), *Shop & Service* (SS), *Residence* (R), *Arts & Entertainment* (AE), and *College & University* (CU). For the data tensor of geo-tagged tweets, we first construct a matrix LT, where LT_{ij} is the number of geo-tagged tweets that fall in the i-th district at the j-th hour in the week. Similarly, we construct the LT matrix based on the traffic sensor data, where LT_{ij} is the average occupancy rate in i-th district at the j-th hour in the week. Then, we construct a matrix FTV, where FTV_{ijk} is the probability of Foursquare check-ins in the k-th venue category that falls in the i-th district at the j-th hour in the week. Thus, for each cell at

[3] N 48° 54′ 32.6118″, E 2° 24′ 33.7104″, N 48° 48′ 56.361″, E 2° 14′ 36.7794″.

a given hour in the week, we know from the matrix FTV the probability distribution of activities over the nine categories. Finally, the entries in the data tensor are computed as:

$$\mathcal{X}_{ijk} = \frac{LT_{ij} \times FTV_{ijk}}{\sum_{ijk} \mathcal{X}_{ijk}}, \tag{3.38}$$

for both $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$. $\boldsymbol{\mathcal{X}}_B$ contains the normalized aggregated values over four weeks between Oct. 16th, 2015 (Friday) and Nov. 12th, 2015, and $\boldsymbol{\mathcal{X}}_A$ is constructed based on the normalized values in the following month, between Nov. 20th, 2015 (Friday) and Dec. 18th, 2015.

In our study we set $\alpha = \beta = 10^{-8}$, $\gamma = 5 \times 10^{-7}$ for social media dataset and $\gamma = 10^{-7}$ for traffic sensor dataset. Finally we set $R = 20$ for both datasets.

3.6.1.3 Results The advantage of *PairFac* is that it aligns the respective components of each tensor which share high similarities. This is realized through the fact that the output of *PairFac* is the mobility components as ranked by their associated discriminative scores, with similar components sharing similar scores. It is therefore straightforward to identify the common patterns as well as those discriminative ones. In the following, we pick two common patterns and one discriminative pattern from each dataset to illustrate the advantage of our proposed *PairFac* method. We first show the patterns from the geo-tagged tweets dataset, followed by the ones from the traffic sensors.

Patterns from social media data: Since the *largest difference* method has been shown to best find the split the patterns as in Section 5.4, we use it to separate the common components and the discriminative components in all case studies. There are 19 pairs of common components with small discriminative scores (M = .032, SD = .034) and one set of discriminative components with discriminative scores as .38 and .39, respectively. Below we show several interesting patterns among them all:

Common Pattern 1. Fig. 11 shows the 3rd component one month before (with discriminative score as .0035) and one month after (with discriminative score as .0038) the Paris attacks. We observe that the patterns from each tensor are virtually identical in all three dimensions. This set of patterns primarily corresponds to the activities in professional places. The time usage of this pattern typically falls during the daytime, although we observe a

Figure 11: Common Pattern from social media data (1). 3*rd* component before the attacks and 3*rd* component after the attacks. Two maps show the probability distribution of check-ins in different neighborhoods of Paris before (right) and after (left), where dark red (right) stands for a higher probability. The bottom-left figure shows the distribution of traffic over the week (24×7), where blue lines represent the distribution before the attacks and the red lines represent the one after. The bottom-right figure features the distribution of check-ins over different types of venues (defined in section 6.1.2).

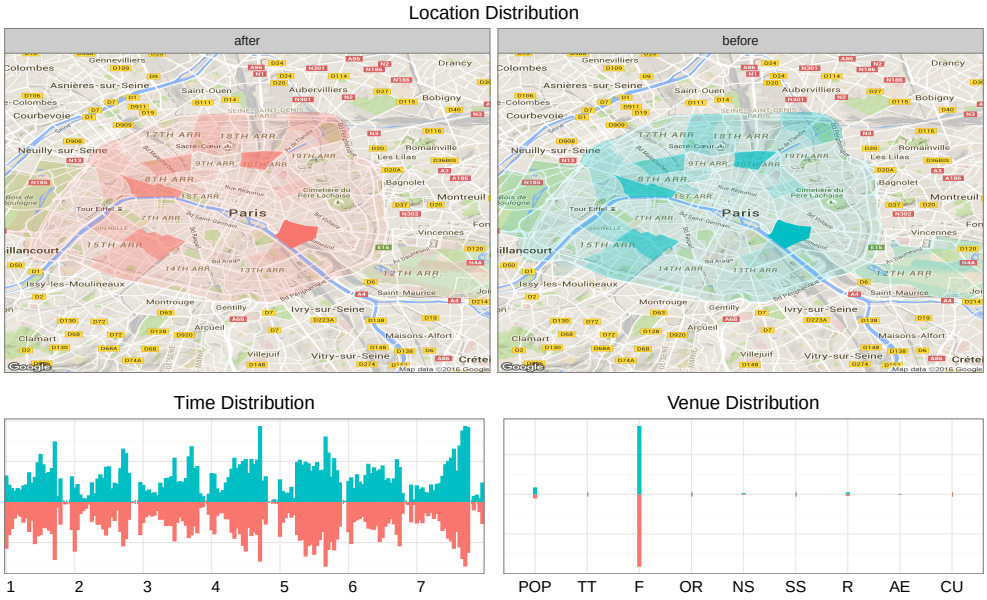

Figure 12: Common pattern from social media data (2). $14th$ component before the attacks and $14th$ component after the attacks.

spike of activities on Thursday nights. This might be due to the small portion of nightlife activities mixed in this pattern. The pattern is heavily geographically distributed in the 16th arrondissement, where four Fortune Global 500 companies (PSA Peugeot Citron, Kering, Lafarge, and Veolia) have their headquarters, which might explain the periodical distribution of professional workplaces activities.

Common Pattern 2. Fig. 12 shows the 14th component one month before (with discriminative score as .026) and one month after (with discriminative score as .053) the attacks. We see that the patterns from each tensor are almost identical, especially in their location and activity distribution. The time associated with this pattern starts from the morning and keeps active for almost the entire day. It is interesting to note that on Sundays, Parisians tend to start this pattern late and then gradually increase its usage. We observe the most geographically highlighted areas are the one that is very close to the 11th arrondissement, which is regarded as the hub of new food scene[4]. Other areas include the 8th and 10th

[4]https://www.thrillist.com/eat/paris/paris-arrondissements-ranked-by-their-food-and-drink

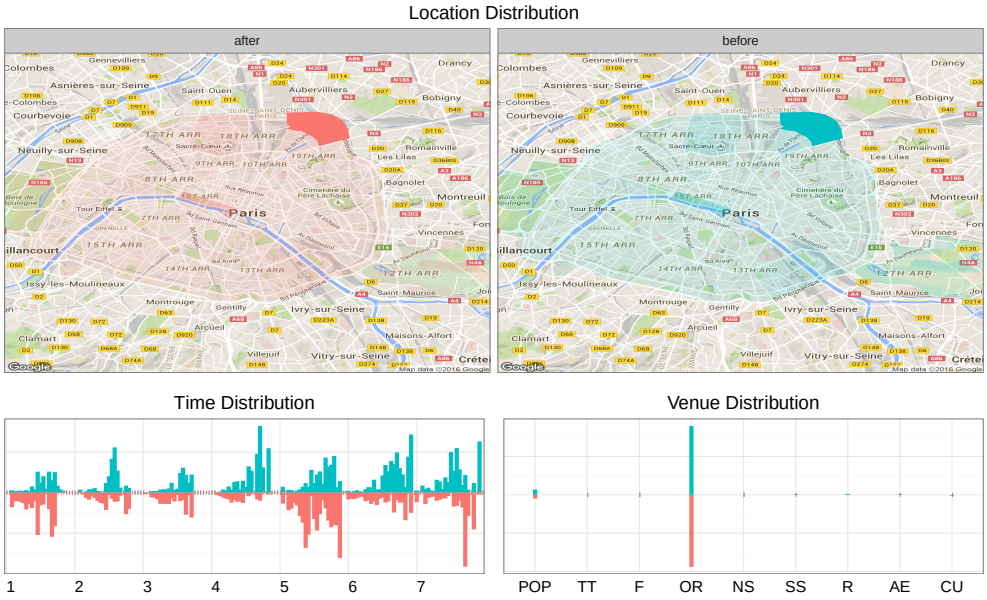

Figure 13: Unique patterns from social media data. 20*th* component before the attacks and 20*th* component after the attacks.

arrondissement are also among the top three popular food places.

Unique Patterns. Fig. 13 shows the 20th component one month before (with discriminative score as .38) and 20th component from one month after (with discriminative score as .039) the attacks. We select these two as they share similar time distribution, along with similar location distribution, while their associated time of the week is different. This set of patterns features the activities around outdoor recreations. The area associated with these components is at the upper corner of 19th arrondissement, which is featured by Parc de la Villette, the third largest park in Paris. This could explain why the activity is centered around the outdoor recreations and the time mostly focuses on the second half of the days or over the weekends. We notice that before the attacks, the time distribution follows a fairly periodical pattern, with activities mostly taking place during the day-time and then shifting to afternoons or nights during the weekends. However, after the attacks, the volume of activities becomes less regular and also shrinks during most of the weekdays.

Patterns from Traffic Sensors: The largest difference method leads to 18 pairs of com-

mon components with small discriminative scores (M = .042, SD = .031) and two sets of discriminative components with large discriminative scores (M = .12, SD = .10), respectively. Below we show several interesting patterns among them all. The first two sets of common patterns are similar to the ones from the social media data in their respective distributions, while the last one differs from each.

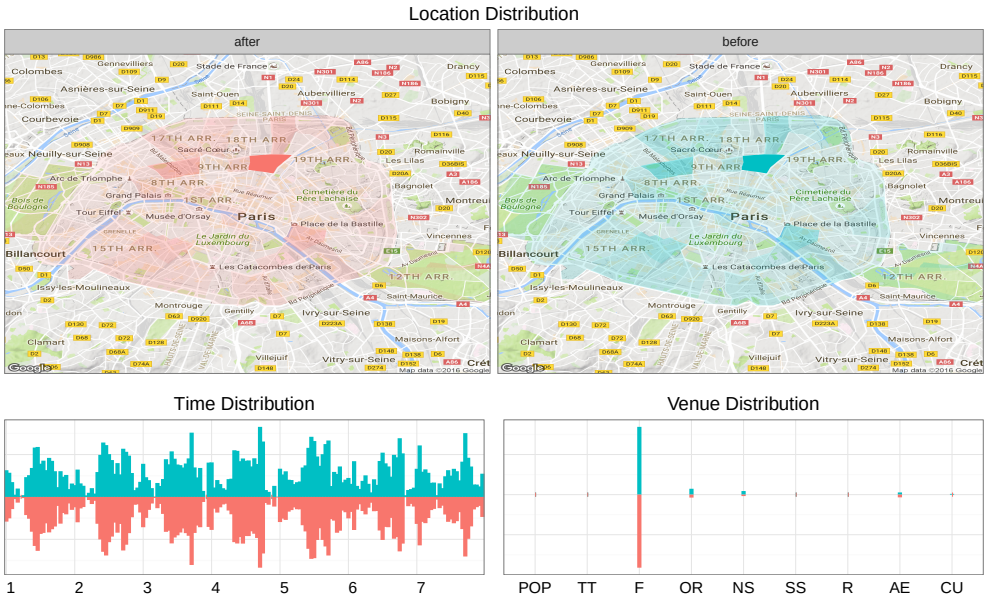

Figure 14: Common Pattern from Paris traffic sensors (1). $6th$ component before the attacks and $6th$ component after the attacks.

Common Pattern 1. Fig. 14 shows the 6th patterns one month before (with discriminative score as .037) and one month after (with discriminative score as .000) the attacks related to the activities of food. We observe that the patterns from each tensor are practically the same in all three dimensions. This set of patterns spans across multiple districts in Paris, while mostly from the 10th arrondissement. In the time dimension, this pattern reaches its peak during the day in the weekdays and tends to peak during the night on the weekends.

Common Pattern 2. Fig.15 shows the 14th components one month before (with discriminative score as .037) and one month after (with discriminative score as .084) the attacks, corresponding to the activities of professional places. The patterns from each tensor are very similar in all three dimensions. This set of patterns spans across multiple districts in Paris.

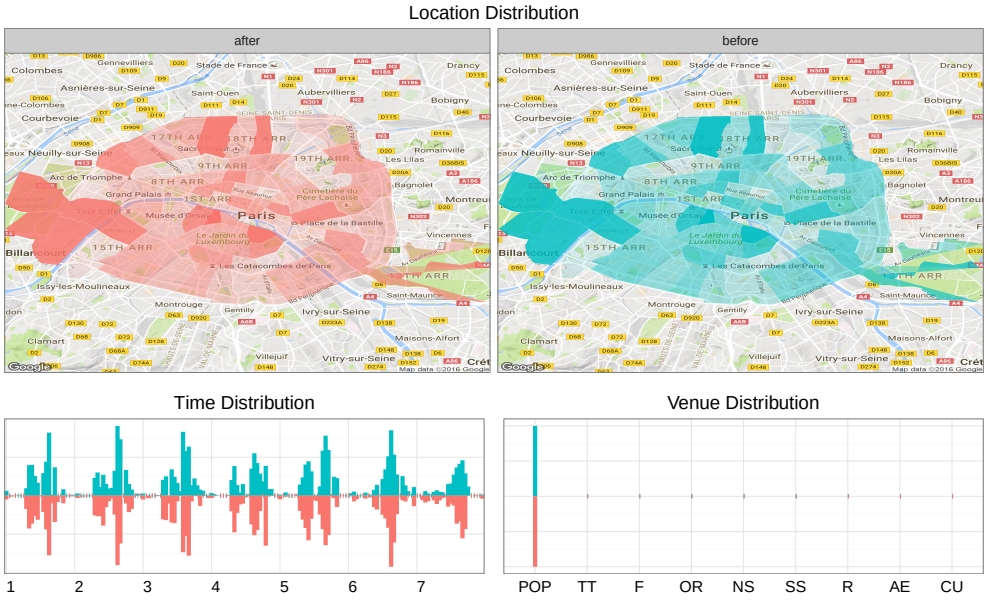

Figure 15: Common pattern 2 from Paris traffic sensors (2). $14th$ component before the attacks and $14th$ component after the attacks.

We can observe two peaks during the day-time, for which we conjecture each of them can relate to the rush hour for work. The weekend traffic, however, is more centralized during the day.

Unique Patterns. Fig. 16 shows the 20th (with discriminative score as .169) component for one month before and 19th (with discriminative score as .096) component for one month after the attacks. We select these two because they exhibit similar distributions both in time and activities, as both of them show daily travel and transportation patterns while being very distinct regarding their location distribution in the city. Prior to the attacks, the destination of travel and transportation seems to fall around multiple locations, while several of them are close the attack sites (e.g., 3rd, 4th, and 11th arrondissement). However, in the following month, the traffic appears to have been more centralized to the 10th arrondissement and also tend to be more spread out from the affected areas. We suspect the difference in the location distribution could be because of road-blocks in those places after the attacks.

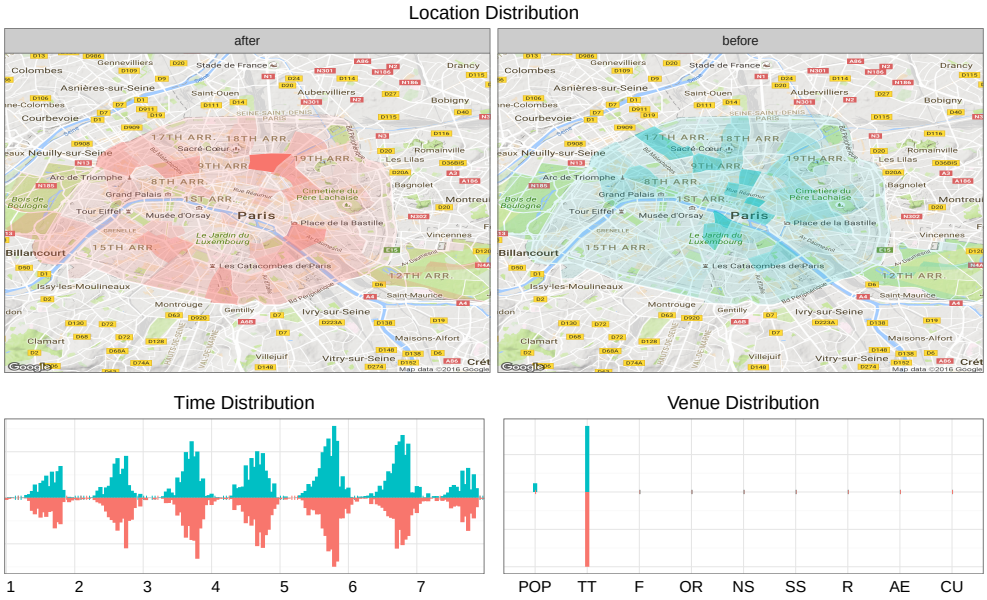

Figure 16: Unique patterns from Paris traffic sensors. $20th$ component before the attacks and $19th$ component after the attacks.

3.6.2 Thanksgiving in NYC

In this section, we demonstrate *PairFac* as a general urban analysis tool to uncover the changes in mobility patterns during holidays. Thanksgiving is a major national holiday in the United States. In this case study, we want to understand the differences in the mobility patterns revealed in the Thanksgiving holiday week over two consecutive years.

3.6.2.1 Dataset We collected the taxi trips during Thanksgiving week of 2014 and 2015, respectively, which accumulates to 4,845,322 trips. Table 5 lists the dataset used in this case study. The information about each trip includes the pick-up location, drop-off location, pick-up time, drop-off time, the number of passengers. Similar to the previous case study, we also supply taxi trips with Foursquare data to model the location-time venue distribution. It contains 554,791 check-ins from 62,120 POIs in the NYC area between April 2012 and September 2013.

Table 5: Data Sources Used in the Case Study of Thanksgiving Holiday Week in NYC.

Type of Data	Dimensions extracted	Volume of raw data extracted
Taxi Trips	Location, Time	4,845,322 Trips
Check-ins and POI database	Location, Time, Activity	554,791 check-ins with 62,120 POI information

3.6.2.2 Case Study Setup Again, we construct three-mode tensors, where the three dimensions are location, time, and venue type, respectively. We keep the time and venue dimension the same as the Paris attacks case study. For the location dimension, we extract 193 neighborhoods in NYC. For the data tensors, we first construct a matrix LT, where LT_{ij} is the total number of passengers that are dropped off in the i-th neighborhood at the j-th hour in the week. Then, we construct a matrix FTV, where FTV_{ijk} is the probability of Foursquare check-ins in the k-th venue category that falls in the i-th neighborhood at the j-th hour in the week. Thus, for each cell at a given hour in the week, we know from the matrix FTV the probability distribution of activities over the nine categories. Finally, the entries in the data tensor are computed following Eq. 3.38. In our experiments, we set $\alpha = \beta = 10^{-8}$, $\gamma = 10^{-7}$, and $R = 10$.

3.6.2.3 Results The largest difference method suggests only one set of discriminative components 10th component before (with discriminative score as .43) and 10th after (with discriminative score as .45), with the rest being common components. However, our observation is that components starting from 8th have already shown different degrees of differences in their distributions. This suggests that using a single cut-off to differentiate common and discriminative components might be too simplified a measure to determine the categories of the components. On the other hand, the discriminative score provided by our model can potentially provide a more accurate measurement of the similarity of the components.

In this section, we show several interesting patterns revealed by our method. Again, we

Component Index	1	2	3	4	5	6	7	8	9	10
Before	$1.1e-02$	$1.2e-02$	0.027	0.00508	0.039	0.021	0.087	0.18	0.19	0.43
After	$2.2e-05$	$1.2e-07$	0.000	0.02174	0.000	0.060	0.098	0.18	0.20	0.45
Difference	$0.0e+00$	$9.8e-04$	0.014	0.00019	0.012	0.042	0.103	0.17	0.04	0.48

Table 6: The discriminative Scores Associated With Each Component in NYC Case Study. The third row shows the difference between the components with the consecutive indexes.

first show two common patterns, followed by two sets of discriminative patterns:

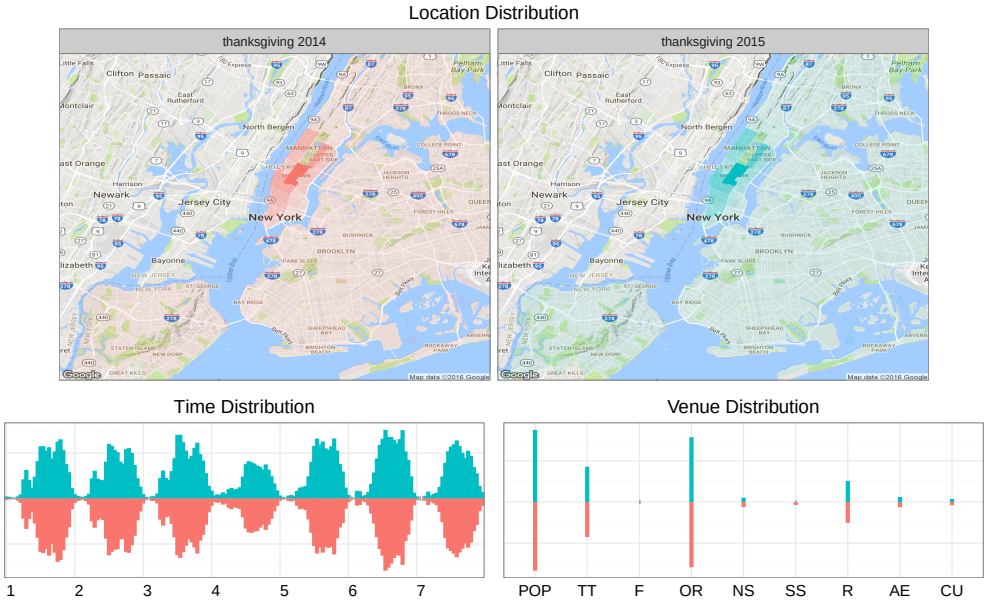

Figure 17: Common patterns from NYC taxi trip (1). 1st component from 2014 Thanksgiving week and 1st component from 2015 Thanksgiving week

Common Pattern 1. Fig. 17 shows the first components from 2014 (with discriminative score as .001) and 2015 (with discriminative score as .000), respectively. Their activity distributions reveal a pattern of mixed functions including professional places, outdoor recreations, travel, and transportation, etc. The activities mostly center during the day-time and the areas associated with this pattern (e.g., Times Square and Central Park) suggest the associated activities (e.g., Times Square for professional places and transportation hubs). We

observe that these two patterns have almost identical distributions in all three dimensions with Thursday (the day of Thanksgiving) being the least active day.

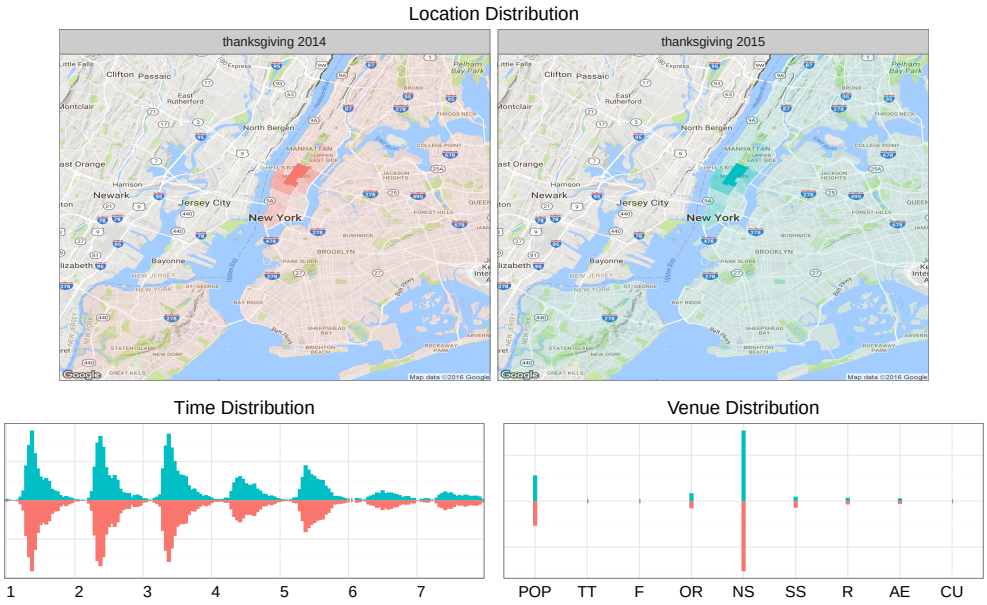

Figure 18: Common patterns from NYC taxi trip (2). 2*nd* component from 2014 Thanksgiving week (red) and 2*nd* component from 2015 Thanksgiving week (blue).

Common Pattern 2. Fig 18 shows the comparison between the 2nd components from Thanksgiving week in 2014 (with discriminative score as .0.001) and 2015 (with discriminative score as .0.000). We observe that the patterns in 2014 and 2015 are almost indifferentiable in all three dimensions. This set of patterns focuses on the nightlife spots activities around Times Square with their peaks spanning from Mondays to Wednesdays while decreasing on.

Unique Patterns 1. Fig.19 shows both 8th component of 2014 (with discriminative score as .18) and 8th component of 2015 (with discriminative score as .18) center their activities around Midtown. However, during the Thanksgiving week of 2014, the focus of the activities from this pattern is related to professional places or colleges and universities, while the focus moves to food and professional places in 2015. For the time dimension, we observe there is a higher volume of activities over the weekend in 2015 and slightly more activities on the Thanksgiving day, comparing to the one in 2014.

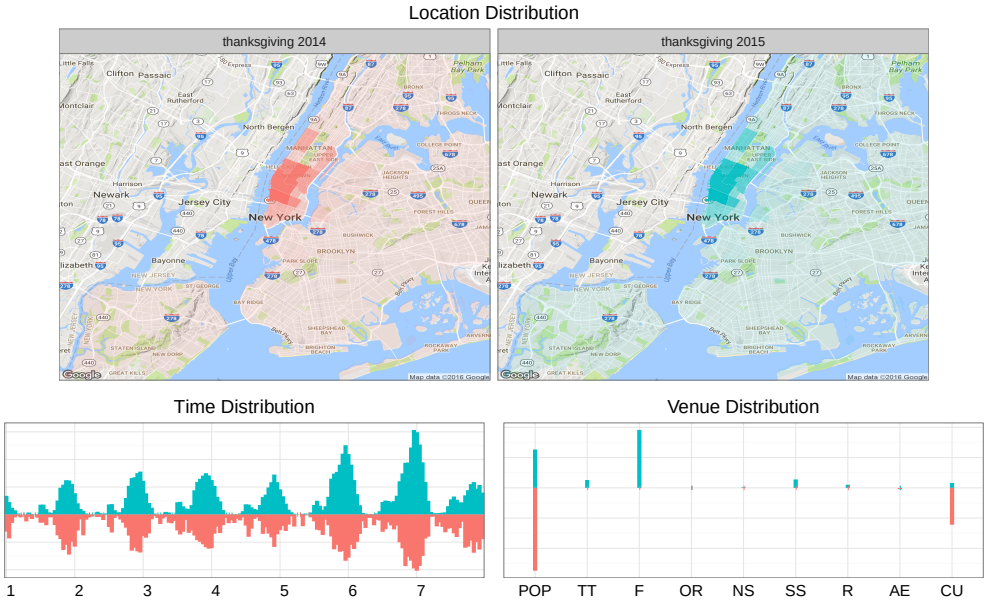

Figure 19: Unique patterns from NYC taxi trips (1). $8th$ component from 2014 Thanksgiving week and $8th$ component from 2015 Thanksgiving week.

Unique Patterns 2. Fig.19 shows both 10th component of 2014 (with discriminative score as .43) and the 10th component of 2015 (with discriminative score as .45) that center their activities around Midtown, LaGuardia, and JFK. Although two patterns have almost identical location preferences, the time and the venue associated differ. While the pattern in 2014 has an array of activities (food, professional places, shopping and services and travel & transportation), the one in 2015 primarily focuses on the shopping and services (e.g., shopping in 5th Ave). In the time mode, the pattern in 2014 has a relative low volume of activities during the weekdays with a higher volume over the weekend, while in 2015 the pattern seems to posses relatively less changes between the weekdays and the weekends. This could be due to the effect of the weather conditions during these two periods of time. In Thanksgiving holiday week 2014, there was a significant winter storm and it wasn't until the weekend that the temperature were back in the high 40's[5]. However, the Thanksgiving

[5]https://www.nbcnewyork.com/news/local/New-York-City-New-Jersey-Snow-Thanksgiving-Travel-Delays-Roads-Forecast-283718461.html

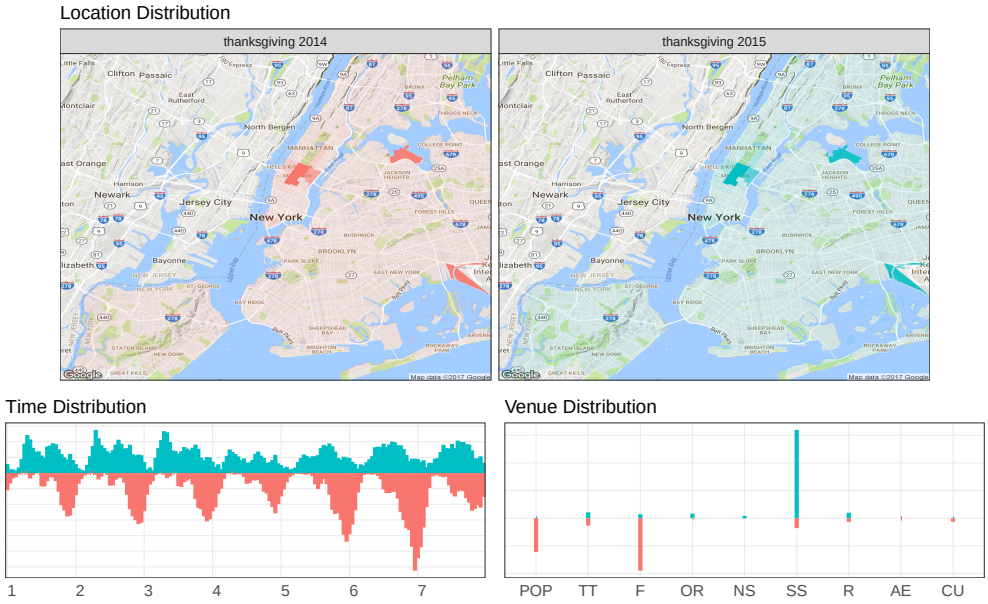

Figure 20: Unique patterns from NYC taxi trips (2). $10th$ component from 2014 Thanksgiving week and $10th$ component from 2015 Thanksgiving week.

holiday week in 2015 has seen sunny weather all week with high temperatures from the upper 40's to the mid-60's.

3.7 DISCUSSION

In what follows we discuss some open issues with our study as well as our future directions:

(1) As the first chapter of this dissertation, we set out to address the mis-match between human information need and the conventional reconstruction-driven factorization from the multi-aspect data. Specifically, we approach through understanding the particular need of information need in the context of impacts of evaluating major events in the urban space and then formally formulating it as a problem of contrastive pattern discovery from a pair of multi-aspect data. In this way, the human information need can be considered as one of optimization goals while optimizing towards the reconstruction of the original multi-aspect

data.

(2) Comparing to the existing work, this study undertook the effort of removing the need to manually pre-determine the number of common and discriminative components. Despite the advancements we made, there still exists the challenging question of the choice of the number components for *PairFac*. Although it is a common question for tensor factorization, and dimensionality reduction tasks in general, it is also an essential step towards a more robust discovery of latent patterns, and the impact of an event in our study in particular. As part of our future work, we plan to investigate a more systematic way of determining the number of components, particularly for the application of event analytics.

(3) *PairFac* is useful in discovering the changes in multidimensional data during two time periods. There are two potential issues with the current model: a) compared to existing literature [122], it does not offer insights on the changes over multiple periods of time; b) the current model focuses on finding the changes in the whole subspace rather than in any particular dimension. The latter could potentially be tackled through a dimension-specific regularization term in *PairFac*'s optimization objective. However, the task of analyzing changes over multiple time-periods could be more challenging, since *PairFac* requires the computation of the auxiliary tensors that host the pair-wise common and discriminant signals. The number of auxiliary tensors needed would increase dramatically as the number of original tensors (i.e., time periods) increases. Hence, more research is required to determine how to scalably model persistent and changing patterns over multiple time periods.

(4) The impact of disasters can be measured from different aspects based on the availability of different datasets. By investigating the disasters using multiple datasets, it is possible to discover the impact that might otherwise be obscured in isolated datasets. The construction of input tensors and the interpretation of the output from the different data sources would depend on the nature of the datasets (i.e., their meanings and granularities). For example, Twitter data contains specific information regarding activities such as locations, times, and content, while sensor data provides broad information about traffic flow as measured by the vehicles. These two datasets provide complementary aspects of human mobility – the kinds of places they visited and tweeted about or how they use vehicles to move around the city. For example, in our previous study of the immediate impact of ur-

ban mobility after the Paris attacks [231], we observed more Twitter activity close to night entertainment areas, but much less traffic. In a scenario of disaster aftermath, Twitter data could help identify how people went out to the streets to show solidarity, or commemorate the victims, whereas the traffic sensor data could show how people's activities on the streets subsequently blocked road segments and reduced the automotive traffic in the same region. As different datasets illuminate distinctive aspects of city dynamics, it is an interesting next step to investigate the correlations among different datasets in order to devise models that can be utilized to discover patterns.

(5) The case studies presented in Section 6 construct tensors with three dimensions (locations, time, venues), where each employs a pre-determined level of details in the corresponding mode revealed from *PairFac*. For example, we use neighborhoods for the location dimension. We proceed to this level because it enables us to further study the potential factors that lead to the observed changes in different neighborhoods. These factors can be obtained from readily available data, such as demographics, which are usually aggregated at the level of city neighborhoods. It is important to note that, with different settings, we might obtain understandings of the urban dynamics in different resolutions. The choice of resolutions at this moment is rather application dependent. In our future work, we aim to develop an extension of our method that can automatically disclose the most interesting details.

(6) Despite the issues and limitations of acknowledged as above, *PairFac* provides interesting insights in evaluating the impact of events in the city. In our first case study of the Paris attacks, we reveal the changes in the mobility patterns based on two datasets, social media data (geo-tagged Twitter content) and traffic sensors, separately. Compared to [231], the results show that most of the patterns resumed to the same orders as they were before the attacks (e.g., Work-related patterns in Fig. 11 and 15, Food-related patterns in Fig. 12 and 14). However, according to the Twitter data, the outdoor-recreation pattern in the northern part of Paris has not been as exercised as much as it was before. Particularly, Thursdays see one of most reduced activities. We guess this might be due to the police raid in the northern suburb of Paris, Saint-Denis, which is close to Parc de la Villette, on November 18th. Although the siege was ended in the morning of November 18th, it wasn't

until the next day that French officials announced the primary suspect in the Paris attacks was killed in the raid [6]. On the other hand, from the traffic sensor data, we show the transportation pattern has seen the distinct focus of regions, where people tend to alternate their choices of transportation to the areas that are away from the attack sites. This change in transportation patterns was only observed from the traffic sensor data. We guess this could be due to the road blocks in the areas close to the attack sites where the access could be limited to foot traffic. The results from the NYC Thanksgiving case study (comparing 2014 to 2015 are contradicting to our expectations as we observe almost identical patterns of Outdoor, Transportation (Fig. 17) and Nightlife activities (Fig. 18), and surprisingly more food activities (Fig. 19). Although FBI has warned that the media officer of ISIS had called the Macys Thanksgiving Day parade an "excellent target," our analysis shows that the mobility patterns do not vary much over the two years even under the influence of potential and imminent terror attack. However, this could also because of reinforced security due to the terror threat as 2,500 police officers were deployed on the ground for the Thanksgiving [7].

3.8 SUMMARY

In this work, we propose a new analytic approach *PairFac* that aims to discover the impact of an exogenous event on multiple aspects of human activities in the urban environment. With the multidimensional nature of the mobility/behavioral data, we formulate the impact discovery as the problem of identifying common and discriminative subspace from these datasets. Compared to the existing methods, our approach has the advantage of automatically distinguishing the common and discriminative components. This is realized through the introduction of auxiliary tensors and additional column regularization for the learning optimization objective of discriminative weights. We conduct extensive experiments with synthetic data to demonstrate *PairFac*'s effectiveness and scalability.

We apply *PairFac* in two case studies and demonstrate its capability to reveal persistent

[6] https://en.wikipedia.org/wiki/2015_Saint-Denis_raid

[7] http://www.nydailynews.com/new-york/hundreds-turn-thanksgiving-parade-balloon-inflation-article-1.2447267

and changing mobility patterns with respect to events of interest. For example, in our first case study using data from the terrorist attacks in Paris of 2015, we see that activities around professional life and food venues experienced the least changes. Using *PairFac* process results, they appear to have identical location and time distribution over the course of the period of study. The most dramatic change was seen in outdoor recreation activities in the 1st and 19th Arrondissements. Although they share the same location distribution, we observe that their associated times became irregular.

PairFac is not only for use in determining disaster impact, as seen in the Pairs terrorist attacks case study, but also as a general urban analysis tool to identify changes in the activities of the city's inhabitants over the different periods of time. This use of *PairFac* can be seen in the example of our second case study. We applied *PairFac* to the data regarding taxi travel during the Thanksgiving holiday week in 2014 and 2015, in order to investigate the changes, if any, in mobility patterns. The results suggest that most of the patterns remain consistent and reveal the unique attributes of mobility in NYC during this major American holiday. For example, in the Times Square area, both nightlife and professional activities decrease between Thanksgiving Thursday through Sunday. When we compare the two Thanksgivings, there are some differences in activities. Specifically, in Thanksgiving 2015, in midtown Manhattan, there are less professional and academic activities, but a greater number of food related activities in the same area with similar time distribution. One potential explanation for the increase in food related activities for Thanksgiving 2015 is that people were not influenced to change their behavior by the conceivable increase in risk of terrorist attack. Additionally, the police took extra precautions and placed heavy police force on the ground to safeguard the areas of the city most at risk[8].

There are several future directions for this work. (1) In this study, we present the case studies from the perspectives of two datasets with distinct nature of their origins and representations separately (e.g., Twitter check-ins and traffic sensors from Paris). In our next step, we also would like to investigate what stands in common for these two data sources. We believe this would shed light on how we could better understand the phenomenon that originates from different data sources. (2) It is also our desire to study how the patterns differ

[8]https://nypost.com/2015/11/26/nypd-beefs-up-security-ahead-of-thanksgiving-day-parade/

and evolve over a period of time instead of only considering before and after the events. We should note that pairwise computation of common and discriminative tensors as introduced in this study make sense for the purpose of probing the shifts during these two periods. However, such design should be used with caution since it could be too computationally expansive for a sequence of tensors over time. In this case, a different design for the computation of the common and discriminative signals might be required. We believe that using the mean of a sequence of tensors could be a more natural way to capture the common signals over time. However, future work is needed to comprehensively understand the problem and to explore potential solutions. (3) Another natural extension of our work is to investigate the driving factors that direct the observed changes. This can be particularly insightful in building disaster impact predictions. (4) As the current output of our algorithm ties to the choice of the number of components, we are not guaranteed to obtain meaningful patterns with a certain designated number of components. To resolve this issue, we want to extend *PairFac* under the framework of hierarchical impact discovery by including a component ranking approach across multiple levels.

4.0 IDISC: ITERATIVE DISCRIMINANT TENSOR FACTORIZATION FOR BEHAVIOR COMPARISON IN MASSIVE OPEN ONLINE COURSES

In the second work, we establish the realization of **Multiplex Pattern Discovery** and **Multifaceted Pattern Evaluation** in the context of understanding the association between latent multi-aspect user behavioral phenomena and performance outcomes in the MOOC platforms. When comparing data structures of users from different performance groups, differences can reside either in high-level or fine-grained patterns. Revealing patterns with a hierarchical structure would add value to the understanding of semantic relationships among the patterns. To this end, we propose a tensor-based learning method, iterative Discriminative tensor factorization *iDisc*, that discovers the common and discriminative learning patterns at multiple levels (Chapter 4.5.1). Besides, *we propose multifaceted pattern evaluation to examine the results of iDisc from the perspectives of pattern validity (Chapter 4.5.2) and utility (Chapter 4.5.3).* We involve experts in a process of manually validating the patterns, followed by a course-end performance prediction task to inspect the utility of the patterns.

4.1 INTRODUCTION

While massive open online courses (MOOCs) have been attracting an ever-increasing number of students, the low completion rate (between 5%-10% [95]) has been a major obstacle to the transformative potentials of MOOCs. The predictive analysis of student performance thus emerged an an important research topic offering insights to platform developers and instructors in arranging proper learning support and allocating resources to students. To

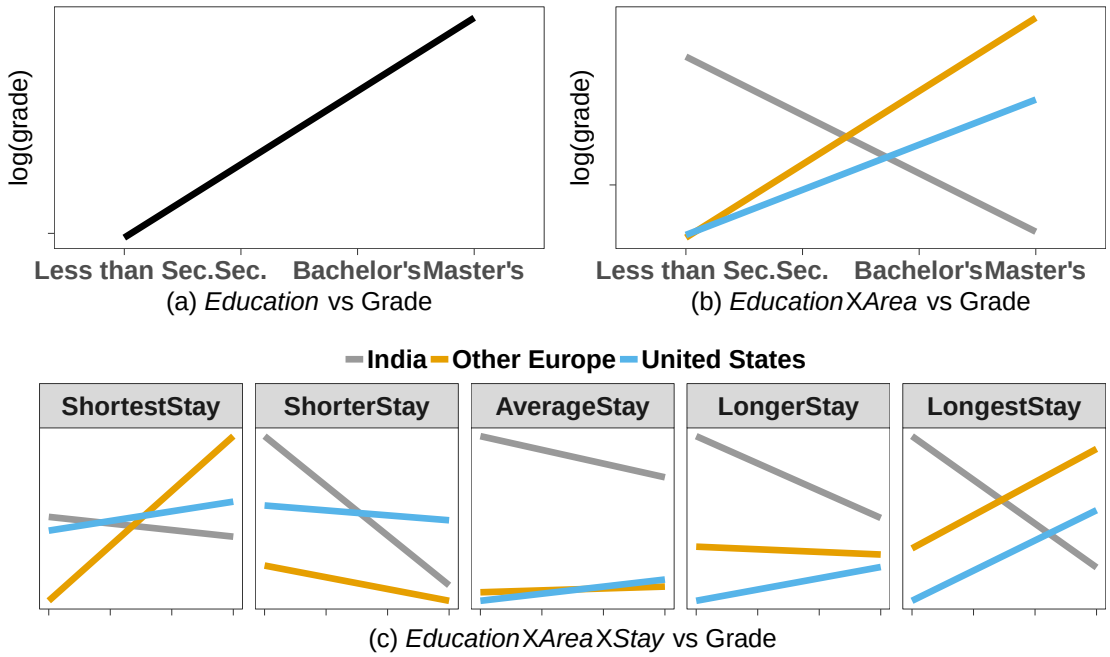

Figure 21: Association Analysis of Student Performance on MOOCs. (a) shows the positive association between the grade and education level; (b) shows the mixed associations when including the area where the student is from; (c) further breaks down the observed groups into different level of activity (five quantiles w.r.t. the number of days students remained active on the platform).

find informative predictors, researchers have focused on extracting features from students interaction with the MOOC platform, such as watching videos, working on assignments, and viewing or contributing to discussion forums. Applied predictive models range from standard machine learning methods [104,151] to more advanced ones such as deep learning [60]. These prediction models could be useful in predicting learning outcomes but are notably limited in helping understand the underlying learning behavior.

There is abundant work that aims to better understand the behavior patterns that relate to the learning outcomes. For example, Coleman et al. [41] correlates each "behavior topic" to the learning outcomes based on topic modeling. However, the research to date fails to consider the multi-dimensional nature of the features and their potential interactions in

outcome learning. Meanwhile, the famous Simpson's paradox points out that the direction of an association at the population-level may be reversed within the subgroups comprising that population [99]. To further explain this in the context of the MOOC platform, we use the Edx MOOC dataset [79] and investigate the factors associated with the students' grade on it.

We extract the education, area (the region where the student is from), the number of active days and the final grade of each student in the course "Introduction to Computer Science and Programming" offered by MITx in Spring 2013. The course had more than 44,000 online participants. Figure 21 shows the association between the selected factors and the final grades of the students. By comparing Figure 21(a) and Figure 21(b), we observe that when the factor of area is considered, mixed associations occur. For example, the subgroup of Indian participants exhibited a negative association between education level and grade, which potentially suggests a more conservative understanding of the relationship between educational background and course outcome. Moreover, when we consider the number of active days, as in Figure 21(c), we notice that for the participants from the "other European" area, the positive association has a strong presence— but only with *ShortestStay* and *LongestStay*. Thus, in comparison to typical correlation analysis, a prediction model that can take advantage of the multi-way interactions of the features could potentially yield better performance.

A growing body of research seeks to resolve Simpsons paradox through causality inference [174]. However, causality is not the focus of this study as we aim for a data-driven approach to subgroup comparisons and explorations. In many cases, this can be interesting and important even in non-causal settings. A straightforward solution is to perform regression with feature interactions, or use Factorization Machines [184] that allow for the estimation of high-order interaction effects. However, the drawback of these methods is that they offer little understanding of the underlying multi-way learning behavior dynamics and their relationship to the learning outcomes. On the other hand, Factorization models like Matrix Factorization (2-way) and Tensor Factorization (m-way) are able to provide an in-depth understanding of meaningful behavior dynamics [67], but there are a few drawbacks preventing them from being more widely adopted by researchers in the field. First,

the associations are isolated, with each of them capturing a certain trend of the behaviors separately (e.g., Figure 22(c)). Second, conventional pattern discovery through factorization models provides little support for contrasting pattern exploration that aims to identify the shared and discriminative behavior characteristics among different groups of users. Being able to do this can tremendously improve knowledge of user behaviors in the context of user group analysis.

In this chapter, we formulate the problem of understanding learning behavior in MOOCs as (1) the simultaneous factorization of the association between students' multi-aspect features and their performance, and (2) the iterative discovery of interpretable shared and discriminative patterns at multiple levels. The critical challenge is how to utilize the multi-way interaction of the features while providing interpretable patterns to help domain experts understand the learning dynamics. We propose a tensor-based learning method— iterative Discriminative tensor factorization (*iDisc*)—that discovers the common and discriminative learning patterns at multiple levels, and based on which we project users to a latent space (i.e. *embedding* for the downstream prediction tasks) to identify the association between the multi-way interaction of the features and the students' performance. To this end, we first represent the behaviors of the students from the opposite performance groups as *coupled tensors*. Since the coarse-grained joint factorization of these behavior tensors may not be capable of revealing behavior patterns at the subgroup level, *iDisc* iteratively performs discriminative pattern discovery at multiple levels. To increase the interpretability of the entire pattern space, we also introduce the inference of pattern hierarchy. To make the solution capable of handling unseen students, we project the students' behavior tensors into a latent space, by considering the multi-way interactions at different levels as the loading matrix. The empirical studies with the dataset from different MOOC platforms have shown the promising results on the effectiveness and efficiency of *iDisc*.

Our contributions can be summarized as follows:

- We formulate the problem of identifying the multi-way feature interaction with interpretable pattern discovery for understanding user behavior on the MOOC platforms.
- We propose a framework of iterative discriminant factorization for multi-way data. By factorizing the residual tensors at each level, our method enables the discovery of com-

mon and discriminative patterns at different granular levels. To ensure the parsimony of the discovered structure, we employ sparse learning to effectively capture enforcing relationships between the top-level and bottom-level patterns.

- We perform extensive experimentation of our methodology using several real-world datasets, and show the efficiency and interpretability of our proposed method.

4.2 RELATED WORK

4.2.1 Predictive Modeling in MOOCs

There are several types of predictive models in MOOCs that are closely related to this work. One direction is to utilize more complex feature types, including higher-order n-gram representations of learner activity data. For instance, features are constructed using the occurrence of pre-defined sequential activities [225], or from sequential pattern mining [73,88, 134,188]. Another line of work proposes to utilize the temporal nature of the activity data for student success prediction. Qiu et al. [180] propose a latent dynamic factor graph (LadFG) to model and predict learning behavior in MOOCs. LadFG captures the dynamic information and homophily correlations between students. It also projects students learning behavior into a latent continuous space for predicting student performance. Another approach is the latent variable modeling as a way of inferring complex relationships between predictors [76,129,139]. For instance, Halawa et al. [76] explore the use of count-based learning activity features to predict dropout; this approach suggests that both observable learner activity and dropout are driven by latent, unobservable "persistence" factors.

4.2.2 Multi-level Pattern Mining

Multi-Level Tensor Factorization addresses the problem of approximating the hierarchical low-rank tensor format. This process allows the representation of the tensors in a nested subspace, in one of Tree-Tucker format [161], tensor train format [160], or tensor networks format [36]. Huang et al. [90] employ a tree-guided learning via tensor decomposition and

matrix factorization in the context of experts recommendation in multiple areas simultaneously. However, there is limited research that discovers hierarchical nested subspace in the tensor subspace [162]. Özdemir et al. [162] construct a data-dependent multi-scale subspace to better represent the data. To do so, the authors first construct a tree structure by partitioning the tensor into a collection of permuted sub-tensors, and then construct the multi-scale subspace by applying HoSVD to each sub-tensor.

Summary. The existing predictive analytics on MOOCs considers various ways of constructing a matrix-based feature space. We argue that tensors could be a more suitable representation for student behavioral modeling due to their flexibility in representing the multi-way interaction of the behavioral data. In this regard, Sahebi et al. [187] have shown success in using a tensor-based approach to model the students' learning process, and predict student performance. However, the multi-way interactions as behavior patterns have not been discussed, and a more interpretable pattern discovery that can support a comprehensive understanding of student behaviors is missing. On the other hand, most hierarchical tensor factorization methods tend to recursively decompose the tensor modes by a pre-specified dimension tree [36, 160]. Our work, instead, is closer to the multi-level tensor factorization approach by [163], which recursively factorizes the residual tensors to obtain a multi-level representation of the subspace. However, to the best of our knowledge, there has been no work yet that discovers the common and discriminative patterns at multiple levels, especially in the area of predictive modeling in educational data mining.

4.3 PROBLEM FORMULATION

In this section, we start with a brief introduction to tensor notions and operations and then formulate the problem considered in this study. Table 1 summarizes the notations used in this chapter.

4.3.1 Problem Formulation

To motivate the problem in the context of a real-world dataset, we discuss the application of NMF, Non-negative Tensor Factorization (NTF), discriminative NTF, and hierarchical NTF with a toy dataset. This dataset was extracted from one of the most popular courses in the XueTangX dataset (full dataset details in Section 5.1). Each student event, or *activity*, is associated with three attributes: *time* (d_1, d_2), *source* (s_1, s_2), and *type* $(t_1, t_2, \ldots, t_7)$.

4.3.1.1 NMF Let matrix $\mathbf{X}$ denote the aggregated activities that users have been recorded engaging in on the XueTangX platform for this course. The nature of matrix $\mathbf{X}$ is a two-dimensional array, which restricts its capability of integrating further information [186]. In this way, we can either drop one of the attributes or force the third dimension to be combined with the second dimension. Figure 22(a) shows the case where $\mathbf{X}'$ contains only *source*×*type*, and Figure 22(b) shows the case where $\mathbf{X}''$ contains *source*×(*type*+*type*), where (*type*+*type*) can be considered as a repeated vector to jointly represent the event activity and the day.

With the behavior described by $\mathbf{X}$, the bottom part of Figure 22(a-b) show the respective low-rank factor matrices approximated by NMF, the *source* factors (left), and the *type* factors (right), since they provide the low-dimensional representations of each source and each activity, respectively. Compared to $\mathbf{X}'$, $\mathbf{X}''$ has the additional advantage of revealing the low-dimensional representation of each activity on different days.

4.3.1.2 NTF Alternatively, we can use a tensor to represent the same dataset (Figure 22(c)). With the given data of a ternary relation nature [210], we could use a third-order tensor $\mathcal{X}$ to denote a *source*×*day*×*type* activity.

NTF techniques can be applied to obtain three low-dimensional representations: *source* factors, *day* factors, and *type* factors, as $\mathcal{X} \approx [\mathbf{S}, \mathbf{D}, \mathbf{T}]$. As a result, each pattern comes with a set: a between-activity vector $\mathbf{t}$ to describe the activity dynamics; a between-source vector $\mathbf{s}$ to describe the usage tendency between different sources; and an across-day vector $\mathbf{d}$ to describe the temporal dynamics (e.g., p_1 in Figure 22(c)). Compared to factorizing the unfolding matrix $\mathbf{X}''$ (Figure 22(b)), NTF introduces the day-specific factors. This sig-

nificantly increases the presentation capability of the patterns by revealing a more direct across-day (temporal) dynamics. Each pattern now represents the interplay of three factors, describing the tendency from different perspectives. With the rich attributes in the behavioral dataset on MOOCs, we thus use tensors to model the behaviors, with the hope that doing so can provide behavioral patterns with the interpretations from different aspects.

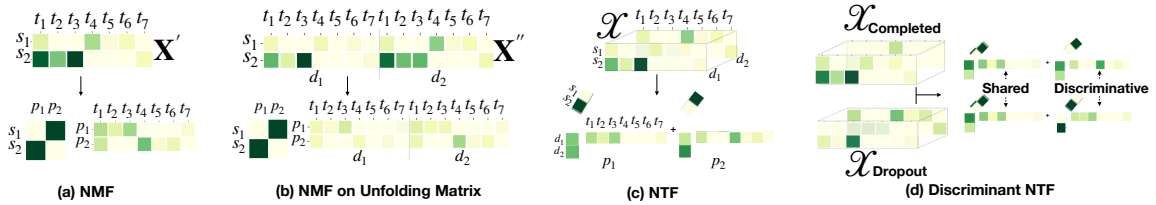

Figure 22: Comparison Between NMF, NMF on Unfolding Matrix, NTF And Discriminant NTF.

4.3.1.3 Discriminant NTF Standard NTF provides meaningful patterns for simultaneously analyzing the behaviors from multiple aspects. This substantially increases the capability of studying and interpreting the behaviors on MOOC platforms with rich dynamics. However, this still does not sufficiently serve the desire to comprehensively understand and investigate student behaviors on these platforms. For example, one of the most interesting questions is which behavior patterns are shared by *completed* students and *dropout* students, and which differentiate the two groups (Figure 22(d)). Through understanding the commonality and differences, researchers can better design course interactions and content to help more students successfully complete the course.

One could easily use NTF to fit the behavior tensors from each group of users, separately. However, this approach does not take advantage of any shared behavior patterns between the two groups. As the behavior moves to high-dimensional tensor space, this could potentially lead to the under-fitting problem. Besides, with patterns generated for each data tensor separately, it needs to perform an additional post-hoc analysis to determine common and discriminative patterns. This is a non-trivial attempt to align the common and discriminative patterns, in the case of each pattern being represented by multiple vectors from different

aspects. In this regard, discriminative NTF is set to jointly factorize the tensors constructed from different groups of users with the following objective considering CP decomposition:

$$\begin{aligned}\mathcal{L}_{\text{disc}} = &\left\|\mathcal{X}_{\text{Completed}} - \left[\mathbf{S}, \mathbf{D}, \mathbf{T}\right]\right\|^2 + \left\|\mathcal{X}_{\text{Dropout}} - \left[\mathbf{S}^{'}, \mathbf{D}^{'}, \mathbf{T}^{'}\right]\right\|^2 \\ &+ \Omega(\mathbf{S}, \mathbf{D}, \mathbf{T}, \mathbf{S}^{'}, \mathbf{D}^{'}, \mathbf{T}^{'}),\end{aligned} \tag{4.1}$$

where $\Omega(\cdot)$ is the function to promote the simultaneous discovery of the common and discriminative patterns [101, 232].

4.3.1.4 Hierarchical NTF

Previous works on NTF or discriminative NTF for unsupervised pattern discovery focus on finding a set of patterns at equal granularity, or in a flat structure. Although they are adequately expressive to reveal the behavior dynamics from different aspects, they can not provide the relations between patterns (such as parent-child and sibling relations).

A hierarchical non-negative tensor factorization (HierNTF) is more desirable than a set of "flat" patterns, because one can work with the pattern exploration in a hierarchy. As opposed to going through each pattern individually, this results in more efficient pattern understanding. HierNTF can be analogous to hierarchical topic modeling, such as hierarchical Latent Dirichlet Allocation (hLDA) [71], where patterns at higher levels in the hierarchy present "abstract" behavior topics, and ones at lower levels reveal more "specific" behavior topics.

4.3.1.5 Problem Statement

Our problem falls into the combination of the discriminative NTF and HierNTF. We would like to identify common and discriminative patterns nested at multiple levels for a deeper understanding of the relationship between students' multi-way behavior dynamics and their course performance. Before we give the formulation of the studied problem, we would like to first clarify some basic concepts used later.

Definition 4.3.1. *Individual Behavior Tensor.* Let $\mathcal{X}^{(u)} \in \mathbb{R}^{I_1 \times I_2 \cdots \times I_M}$ be an M-way tensor representing an individual user u, with each entry in the $\mathcal{X}^{(u)}$ being an activity from user u that is jointly described by M attributes.

The attributes can be data or platform dependent, such as a time-varying attribute tensor constructed from demographic and behavior attributes associated with users at different time stamps [180]. The individual behavior tensor can be considered as a multi-way representation of each student. With that, for each performance group, we can compute the *collective behavior tensor*.

Definition 4.3.2. *Collective Behavior Tensor.* Let $\mathcal{X}_c \in \mathbb{R}^{I_1 \times I_2 \cdots \times I_M}$ be an M-way tensor that users M attributes to describe the collective activities from a group of users indexed by c.

Students at each performance group can be jointly represented by a collective behavior tensor that captures the full multi-way feature interactions of their activities. Then, we can combine the tensors for the two opposite performance groups to construct the *coupled tensors*.

Definition 4.3.3. *Coupled Tensors.* Coupled Tensors $\boldsymbol{\mathcal{X}} = \{\mathcal{X}_c\}$ is a pair of tensors with identical attributes, i.e., $I_1^c = I_1^{\bar{c}}, I_2^c = I_2^{\bar{c}}, \ldots, I_M^c = I_M^{\bar{c}}$, where c is the index of user group that tensor $\mathcal{X}$ is constructed from and $\bar{c}$ represents the counterpart class.

The coupled tensors $\boldsymbol{\mathcal{X}}$ can be constructed in various ways, depending on the performance metric selected, such as $c \in \{\text{dropouts, completion}\}$ or $c \in \{\text{certificates, no-certificates}\}$. Inspired by [178], we construct the coupled tensor as follows:

$$\mathcal{X}_c = \frac{1}{|U_c|} \sum_{u \in U_c} \mathcal{X}^{(u)}, \tag{4.2}$$

where U_c is the subset of users u with u belonging to class c. While each individual behavior tensor captures the full-order feature interactions explained by her activities, the full interplay between the M attributes for each group of students can be contained within the tensor structure corresponding to the group.

Definition 4.3.4. *Multi-way Behavior Pattern.* A multi-way behavior pattern is a a collection of M vectors $(x^{(1)}, x^{(2)}, \ldots, x^{(M)}), M \geq 2$, where $x^{(m)} \in \mathbb{R}^{I_m}$ is a vector to describe the pattern with the m-th attribute.

Definition 4.3.5. *Pattern Hierarchy.* Let $\mathbf{P}^l \in \mathbb{R}^{R_l \times R_{l-1}}$ denote a pattern hierarchy that specifies the relationships between the behavior patterns in the consecutive levels, i.e., level l and level $l-1$, where $1 \leq l \leq L$. $\mathbf{P}^l$ can be considered as a projection matrix that maps the pattern from the l-th level to the ones at the $(l-1)$-th level.

Definition 4.3.6. *Tensor-based User Embedding.* A tensor-based user embedding $v \in \mathbb{R}^R$ is the student's vector representation that preserves each student's behavior tensor with a lower-dimensional feature space $\mathbb{R}^R$, given the tensor's rank R.

With the definitions above, we define the iterative common and discriminative pattern analysis as follows:

Problem 1. Given a set of individual behavior tensors $\mathcal{X}^{(u)} \in \mathbb{R}^{I_1 \times I_2 \cdots \times I_M}, (u = 1, 2, \ldots, N)$ corresponding to C categories, $C = 2$, and a set of unseen test data $\mathcal{X}^{(t)} \in \mathbb{R}^{I_1 \times I_2 \cdots \times I_M}, (t = 1, 2, \ldots, T)$, our goal is to iteratively identify a set of patterns that reveal the common and discriminative behaviors at multiple levels, and then use the learned patterns as bases to infer the user embeddings for prediction of the group membership from the unseen students.

Specifically, the task is threefold: (1) collective behavior pattern inference for discovering the common and discriminative patterns; (2) iterative pattern discovery at multiple levels with hierarchy; and (3) embedding projection for test samples based on the iterative patterns for classification. In other words, the first two tasks are aimed at revealing interpretable patterns that could explain the interplays between the behavior attributes, and the last task is to discover the relationship between student performance and the multi-way patterns.

4.4 SOLUTIONS

In this section, we introduce an iterative tensor factorization method named *iDisc* for the coupled tensors, $\boldsymbol{\mathcal{X}} = \{\mathcal{X}_c\}$. Figure 23 illustrates the overview of *iDisc*. There are two-stages: (1) iterative application of discriminative tensor subspace learning; and (2) representation learning for the unseen student based on the multi-level patterns.

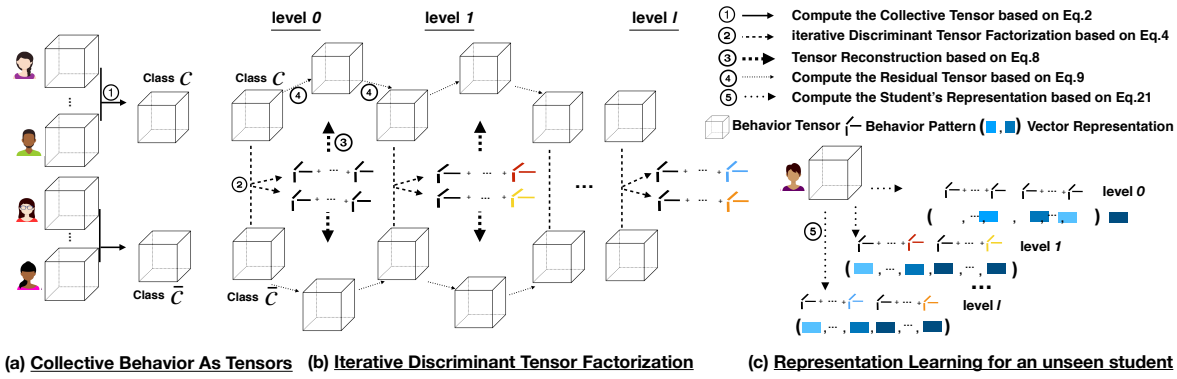

Figure 23: The overview of *iDisc*'s workflow. The workflow consists of three steps: (a) converts behaviors from students of certain performance group to tensors; (2) perform iterative discriminant tensor factorization to identify the shared and unshared patterns at multiple levels; (3) given a new student, transform her behavior to a latent representation based on the patterns identified.

4.4.1 Iterative Discriminative Tensor Subspace Learning

This component iteratively applies the following two-step approach: (1) discriminant low-rank tensor approximation, followed by (2) computing and passing the residual tensor into the next level.

4.4.1.1 Discriminant Tensor Factorization Conventional tensor factorization seeks a set of N factor matrices $[\![\mathbb{U}]\!] = [\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}]\!]_c^l$ from a behavior tensor $\mathcal{X}_c^l$ at level l for class c. One such example is:

$$\mathcal{L}^l = \underbrace{\sum_c \left\| \mathcal{X}_c^{\,l} - [\![\mathbb{U}]\!]_c^l \right\|^2}_{\text{Loss for Coupled Tensors Factorization}} . \tag{4.3}$$

Through Eq. 5.1, we could obtain a set of independent factor matrices (or behavior patterns) for each of the performance groups, respectively. However, this does not consider the commonality and differences among the coupled tensors.

In order to take the commonality between the two behavior tensors into consideration and allow discrimination against each other, we introduce two sets of auxiliary tensors $\mathcal{S}_c^l$ and $\mathcal{Z}_c^l$ that capture the shared behavior and the discriminative behaviors among the coupled tensors. Inspired by [232], $\mathcal{S}_c^l$ and $\mathcal{Z}_c^l$ are computed based on the coupled tensors $\boldsymbol{\mathcal{X}}$ with the clamping function (Eq. 7 and Eq. 9 in [232]). The rational behind the auxiliary tensors is that; we would like to have discriminative tensors that contain only the unique signals for each class, and common tensors to hold what is shared among the coupled tensors. A collective tensor factorization framework is then leveraged to jointly factorize the coupled tensors and the auxiliary tensors as follows:

$$\begin{aligned}\mathcal{J}^l = \mathcal{L}^l + \underbrace{\lambda_0\Big(\sum_c \Big\|\mathcal{Z}_c{}^l - \big[\mathcal{W}_{\mathcal{Z}}; \mathbb{U}\big]_c^l\Big\|^2 + \sum_c \Big\|\mathcal{S}_c{}^l - \big[\mathcal{W}_{\mathcal{S}}; \mathbb{U}\big]_c^l\Big\|^2\Big)}_{\text{Loss for Auxiliary Tensor Factorization}} \\ + \underbrace{f\big(\mathbf{U}_c^l, \mathbf{U}_{\bar{c}}^l, \mathcal{W}_{\mathcal{S}}^l\big)}_{\text{Loss for Pattern Alignment}} + \underbrace{g\big(\mathbf{U}_c^{l-1}, \mathbf{U}_c^l, \mathbf{P}^l\big)}_{\text{Loss for Pattern Hierarchy}} + \underbrace{h\big(\mathbf{P}^l\big)}_{\text{L1 penalty}} \\ s.t. \ \big\|\mathbf{U}_c^l\big\|_2 = 1, \forall\ c,\end{aligned} \tag{4.4}$$

where:

- $\mathcal{W}_{\mathcal{Z}}$ and $\mathcal{W}_{\mathcal{S}}$ are the core tensors with super diagonal entries;
- $f\big(\cdot\big)$ is the function to enforce the similar components to be aligned correspondingly and defined as:

$$f\big(\mathbf{U}_c^l, \mathbf{U}_{\bar{c}}^l, \mathcal{W}_{\mathcal{S}}^l\big) = \lambda_1 \sum_m \Bigg(\Big\|\text{diag}\big(\mathcal{W}_{\mathcal{S}q}\big)\mathbf{U}_c^m - \text{diag}\big(\mathcal{W}_{\mathcal{S}\bar{c}}\big)\mathbf{U}_{\bar{c}}^m\Big\|^2\Bigg); \tag{4.5}$$

- $g\big(\cdot\big)$ is the function that learns the shared mapping $\mathbf{P}^l$ by the coupled tensors, between the patterns at the consecutive levels. We can consider this operation as performing a matrix decomposition from $\mathbf{U}_c^{l-1}$ to $\mathbf{U}_c^l$ and $\mathbf{P}^l$ as:

$$g\big(\mathbf{U}_c^{l-1}, \mathbf{U}_c^l, \mathbf{P}^l\big) = \lambda_2 \left\|\mathbf{U}_c^{l-1} - \mathbf{U}_c^l\mathbf{P}^l\right\|^2, \tag{4.6}$$

  assuming that we already have the values of $\mathbf{U}_c^{l-1}$ for l-th level pattern discovery;

- $h\big(\,\cdot\,\big)$ is an $L1$ penalty function that encourages sparsity in $\mathbf{P}^l$ to promote the exclusive mapping between the factor matrices at the consecutive levels. Considering a more interpretable pattern hierarchy, we use $L1$-norm, since it can function as a proxy for the $L0$ norm, to minimize the number of nonzero elements while maintaining the convexity of the cost function when estimating P the others fixed. In this manner, we ensure the higher-level patterns are mapped to exclusive lower-level ones; and
- λ_0, λ_1, λ_2, λ_3 are the respective parameters to weigh in each objective.

With Eq. 4.4, common and discriminative patterns discovery at the l-th level becomes an optimization objective as:

$$\theta^l = \text{argmin}_\theta^l \; \mathcal{J}^l, \tag{4.7}$$

where $\theta^l = \{\mathbb{U}, \mathcal{W}_\mathcal{Z}, \mathcal{W}_\mathcal{S}, \mathbf{P}\}_c^l$.

4.4.1.2 Obtain the residual tensors Once θ^l is determined, the reconstructed tensor can be obtained by:

$$\hat{\mathcal{X}}_c^l \approx \big[\mathbb{U}_c^l\big], \tag{4.8}$$

and therefore the residual tensor can be computed as:

$$\mathcal{E}_c^l = \mathcal{X}_c^l - \hat{\mathcal{X}}_c^l, \forall\; c. \tag{4.9}$$

Let $\mathcal{X}_c^{l+1}$ denote the tensors for the identification of common and discriminative patterns at the next level $l+1$. We first obtain $\mathcal{X}_c^{l+1}$ as: $\mathcal{X}_c^{l+1} = \mathcal{E}_c^l$, where $\mathcal{E}_c^l$ is the residual tensor as aforementioned. With $\mathcal{X}_c^{l+1}$, we can further identify the common and discriminative patterns at level $l+1$ with Eq. 4.4.

ALGORITHM 2: `iDisc` algorithm for discovering the shared and discriminative subspace from tensor pairs with multiple resolutions.

Input : original tensors $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$, and $\mathbf{R} = \{\}$.

Output: $\{w_q\}$, $\{U_q^{(m)}\}$ for $q \in \{A, B\}$ and $m \in \{L, V, T\}$

Compute $\{\mathbf{U}_q^{(m)}, \forall q \text{ and } \forall m\}_0$ by `PairFac` with $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$, and $\mathbf{R}[0]$;

Compute $\{\{\mathbf{U}_q^{(m)}, \forall q \text{ and } \forall m\}_i | i \in \{\mathbf{R}_i\}, i > 0\}$ by Collective Matrix Factorization;

while *not converged* **do**

  $k = k + 1$;

  Compute $\mathcal{L}_{\mathbf{w}_q}^{k-1}$, $\mathcal{L}_{\mathbf{U}_q^{(m)}}^{k-1}$, and set ω^{k-1}, $\forall q$ and $\forall m$, according to Eq. 3.18, 3.28, 3.19;

  Compute $\hat{\mathbf{U}}_{q,k}^{(m)}$ and $\hat{\mathbf{w}}_{q,k}$, $\forall q$ and $\forall m$, according to Eq. 3.21, and 3.25;

  Update $\mathbf{U}_{q,k}^{(m)}$ and $\mathbf{w}_{q,k}$, $\forall q$ and $\forall m$, according to Eq. 3.23, and 3.29;

end

4.4.1.3 Parameter Optimization of *iDisc*

For simplicity of notation, we omit the level l in all notations since the optimization is performed per level with the focus on the unknown θ^l and the θ^{l-1} are learned at level $l-1$. We also omit the mode notation m because all modes share the same optimization process. Let $\mathbf{U}_c$ represent the mode-m factor matrix at level-l, instead of the rather complex form of $\{\mathbf{U}_c^{(m)}\}_l$. We use $\overline{\mathbb{U}}_c$ to denote the set of factor matrices for $\mathcal{X}_c$ that correspond to modes other than m, and $\bar{c}$ to denote the class that is not c.

Since objective function $\mathcal{J}$ is not convex with respect to θ, we aim to find a local minimum for $\mathcal{J}$ by iteratively updating each in θ.

1. Update $\mathbf{U}_c$*, fix others.* The optimization of $\mathbf{U}_c$ is equivalent to the following least squares loss functions [232]:

$$\begin{aligned}\mathbf{U}_c \leftarrow \underset{\mathbf{U}_c \geq 0}{\operatorname{argmin}} \frac{1}{2}\Big(\frac{1}{n_c}\left\|\mathbf{X}_c - \mathbf{U}_c\,(\odot\overline{\mathbb{U}}_c)^T\right\|_F^2\Big) \\ + \lambda_0\Big(\left\|\mathbf{Z}_c - \mathbf{U}_c\Lambda_{\mathbf{w}_{Z_c}}(\ \odot\overline{\mathbb{U}}_c)^T\right\|_F^2 + \left\|\mathbf{S}_c - \mathbf{U}_c\Lambda_{\mathbf{w}_{S_c}}\,(\odot\overline{\mathbb{U}}_c)^T\right\|_F^2\Big) \\ + \lambda_1\left\|\mathbf{U}_c\Lambda_{\mathbf{w}_{S_c}} - \mathbf{U}_{\bar{c}}\Lambda_{\mathbf{w}_{S_{\bar{c}}}}\right\|_F^2 + \lambda_2\left\|\mathbf{U}_c^{l-1} - \mathbf{U}_c\mathbf{P}\right\|^2,\end{aligned} \tag{4.10}$$

where $\mathbf{X}_c$ is the mode-m unfolding of tensor $\mathcal{X}_c$. Then the gradient update of $\mathbf{U}_c$ can be computed as:

$$\begin{aligned}\bigtriangledown_{\mathbf{U}_c}\mathcal{J} = &\frac{1}{n_c}\left(\mathbf{U}_c(\odot\overline{\mathbb{U}}_c)^T - \mathbf{X}_c\right)(\odot\overline{\mathbb{U}}_c) \\ &+ \lambda_0\Big(\left(\mathbf{U}_c\Lambda_{\mathbf{w}_{Z_c}}(\odot\overline{\mathbb{U}}_c)^T - \mathbf{Z}_c\right)(\odot\overline{\mathbb{U}}_c)\Lambda_{\mathbf{w}_{Z_c}}^T + \left(\mathbf{U}_c\Lambda_{\mathbf{w}_{S_c}}(\odot\overline{\mathbb{U}}_c)^T - \mathbf{S}_c\right)(\odot\overline{\mathbb{U}}_c)\Lambda_{\mathbf{w}_{S_c}}^T\Big) \\ &+ \lambda_1\left(\mathbf{U}_c\Lambda_{\mathbf{w}_{S_c}} - \mathbf{U}_{\overline{c}}\Lambda_{\mathbf{w}^{S_{\overline{c}}}}\right)\Lambda_{\mathbf{w}_c} - \lambda_2\mathbf{P}\left(\mathbf{U}_c^{l-1} - \mathbf{U}_c\mathbf{P}\right).\end{aligned} \tag{4.11}$$

2. Update $\mathcal{W}_{Z_c}$, fix others.

Let $\mathbf{w}_{Z_c}$ denote $\mathbf{w}$ for tensor $\mathcal{X}_c$. The optimization of $\mathbf{w}_{Z_c}$ is equivalent to the following problem:

$$\mathbf{w}_{Z_c} \leftarrow \underset{\mathbf{w}_c\geq 0}{\operatorname{argmin}}\ \lambda_0\left\|\mathbf{Z}_c - \Lambda_{\mathbf{w}_{Z_c}}(\odot\mathbb{U}_c)^T\right\|^2, \tag{4.12}$$

where $\Lambda_{\mathbf{w}_{Z_c}} \in \mathbb{R}^{R\times R\times R}$ is the tensor with $\mathbf{w}_{Z_c}$ as its super-diagonal entries. The gradient update of $\mathbf{w}_c$ is:

$$\bigtriangledown_{\mathbf{w}_{Z_c}}\mathcal{J} = \lambda_0\left(\Lambda_{\mathbf{w}_{Z_c}}(\odot\mathbb{U}_c)^T - \mathcal{Z}_c\right)(\odot\mathbb{U}_c). \tag{4.13}$$

3. Update $\mathcal{W}_{S_c}$, fix others.

Let $\mathbf{w}_{S_c}$ denote $\mathbf{w}$ for tensor $\mathcal{X}_c$. The optimization of $\mathbf{w}_{S_c}$ is equivalent to the following problem:

$$\mathbf{w}_{S_c} \leftarrow \underset{\mathbf{w}_{S_c}\geq 0}{\operatorname{argmin}}\ \lambda_0\left\|\mathbf{S}_c - \Lambda_{\mathbf{w}_{S_c}}(\odot\mathbb{U}_c)^T\right\|^2 + \lambda_1\left\|\mathbf{U}_c\Lambda_{\mathbf{w}_{S_c}} - \mathbf{U}_{\overline{c}}\Lambda_{\mathbf{w}_{S_{\overline{c}}}}\right\|^2. \tag{4.14}$$

The gradient update of $\mathbf{w}_{S_c}$ can be derived as:

$$\bigtriangledown_{\mathbf{w}_{S_c}}\mathcal{J} = \lambda_0\left(\Lambda_{\mathbf{w}_{S_c}}(\odot\mathbb{U}_c)^T - \mathcal{S}_c\right)(\odot\mathbb{U}_c) - \lambda_1\left(\mathbf{U}_c{}^T\left(\Lambda_{\mathbf{w}_{S_c}}\mathbf{U}_c - \Lambda_{\mathbf{w}_{S_{\overline{c}}}}\mathbf{U}_{\overline{c}}\right)\right) \tag{4.15}$$

4. Update $\mathbf{P}$, fix others. The optimization of $\mathbf{P}$ is equivalent to a co-regularized collective matrix factorization problem with sparsity constraints [53, 198]:

$$\mathbf{P} \leftarrow \underset{\mathbf{P}\geq 0}{\operatorname{argmin}}\ \lambda_2\big(\frac{1}{2}\left\|\mathbf{U}_c^{l-1} - \mathbf{U}_c\mathbf{P}\right\|^2 + \frac{1}{2}\left\|\mathbf{U}_{\overline{c}}^{l-1} - \mathbf{U}_{\overline{c}}\mathbf{P}\right\|^2\big) + \lambda_3\left\|\mathbf{P}\right\|_1. \tag{4.16}$$

Since the sparsity is applied to each row of $\mathbf{P}$, each row $\mathbf{P}_{(r,:)}$ can be updated based on the following gradient:

$$\begin{aligned}\bigtriangledown_{\mathbf{P}_{(r,:)}}\mathcal{J} = -\lambda_2\big(\mathbf{U}_{c\ (r,:)}^{T}(\mathbf{U}_{c}^{l-1} - \mathbf{U}_{c(:,r)}\mathbf{P}_{(r,:)}) \\ + \mathbf{U}_{\overline{c}\ (r,:)}^{T}(\mathbf{U}_{\overline{c}}^{l-1} - \mathbf{U}_{\overline{c}(:,r)}\mathbf{P}_{(r,:)})\big) + \lambda_3 \text{sgn}(\mathbf{P}_{(r,:)}).\end{aligned} \tag{4.17}$$

The details of *iDisc* are summarized in Alg. 2.

4.4.1.4 Time Complexity Analysis The time complexity is mainly consumed by updating each factor matrix $\mathbf{U}_c$ in *iDisc* from computing $\bigtriangledown_{\mathbf{U}_c}\mathcal{J}$. From Eq. 5.8, we need to compute $\mathbf{U}_c(\odot\overline{\mathbb{U}}_c)^T(\odot\overline{\mathbb{U}}_c)$ and $\mathbf{X}_c(\odot\overline{\mathbb{U}}_c)$ in the first term. Note $\mathbf{U}_c \in \mathbb{R}^{I_m \times R}$ and $\mathbf{X}_c \in \mathbb{R}^{I_m \times \prod_{i\neq m} I_i}$ and therefore we have $\mathbf{U}_c(\odot\overline{\mathbb{U}}_c)^T(\odot\overline{\mathbb{U}}_c) \in \mathbb{R}^{I_m \times R}$ and $\mathbf{X}_c(\odot\overline{\mathbb{U}}_c) \in \mathbb{R}^{I_m \times R}$. The operation of matricized tensor times Khatri-Rao product ($\mathbf{X}_c(\odot\overline{\mathbb{U}}_c)$) is often considered a bottleneck for CP decomposition due to the expensive computational cost [234]. In practice, the sparsity of the tensor is leveraged for an efficient computation for this operation [34, 234]. Particularly, the complexity can be reduced by only considering the computation for nonzero observations in $\mathcal{X}_c$. Let x_h denote the h-th nonzero observation in $\mathcal{X}_c$ and its subscripts in $\mathcal{X}_c$ as $(I_{1_h}, I_{2_h}, \ldots, I_{M_h})$. If there are H non-zeros, i.g, $H = nnz(\mathcal{X}_c)$, we would just need an H-vector to store the real values of $\mathcal{X}_c$. In this case, the element-wise computation for $\mathbf{X}_c(\odot\overline{\mathbb{U}}_c)$ can be written as:

$$\begin{gathered}\big(\mathbf{X}_c(\odot\overline{\mathbb{U}}_c)\big)_{(i,r)} = \sum_{\substack{h=1 \\ I_{m_h}=i}}^{H} x_h \prod_{\substack{m'=1 \\ m'\neq m}}^{M} \mathbf{U}^{(m')}_{(I_{m'_h},r)}\ , \\ \text{for } i = 1, \ldots, I_m \text{ , and } r = 1, \ldots, R,\end{gathered} \tag{4.18}$$

where the computation of Khatri-Rao product can be ignored when $x_h = 0$. Therefore, the time complexity for computing $\mathbf{X}_c(\odot\overline{\mathbb{U}}_c)$ for each mode per iteration is $\mathcal{O}(nnz(\mathcal{X}_c)I_m R)$. The element-wise of $\mathbf{U}_c(\odot\overline{\mathbb{U}}_c)^T(\odot\overline{\mathbb{U}}_c)$ can be efficiently computed as [2, 199]:

$$\big(\mathbf{U}_c(\odot\overline{\mathbb{U}}_c)^T(\odot\overline{\mathbb{U}}_c)\big)_{(i,r)} = \sum_{j=1}^{R}\left(\mathbf{U}^{(m)}_{(i,j)} \prod_{\substack{m'=1 \\ m'\neq m}}^{M} \sum_{i=1}^{I_{m'}} \mathbf{U}^{(m')^T}_{(i,j)}\ \mathbf{U}^{(m')}_{(i,r)}\right). \tag{4.19}$$

Therefore, the time complexity as $\mathcal{O}(\hat{I}\ R^2)$, where $\hat{I} = \sum_{m'=1}^{M} I_{m'} - I_m$ and the overall time complexity for the above two terms is $\mathcal{O}(HI_mR + \hat{I}\ R^2)$. Similarly, the time complexity for terms involving tensors $\mathcal{Z}$ and $\mathcal{S}$ becomes $\mathcal{O}(H^{'}I_mR^2 + \hat{I}\ R^3)$ due to the additional loop introduced by the weight vector $\mathbf{w}$, where $H^{'}$ is the respective nonzero observations in the auxiliary tensors. Since $H^{'} \leq H$, with $M \ll H$, $R \ll H$, and $\hat{I} \ll H$, we can see that the running time is expected to scale linearly with the number of nonzero observations in $\mathcal{X}_c$.

4.4.2 Embedding Learning for the Unseen Student

This section explains the inference of the student's embedding in the latent space anchored by the factor matrices. The individual behavior tensor $\mathcal{X}_t$ for unseen student t with an unknown class is constructed based on his or her logs with the system. The mode settings of tensor $\mathcal{X}_t$ are the same as for the collective behavior tensors (e.g., $\mathcal{X}_c$). With the students' individual tensor and the factor matrices, we follow *iDisc* to first obtain the corresponding auxiliary $\mathcal{Z}$ tensors at l-th level, $\mathcal{Z}^l_{t_{\hat{c}}}$, $\forall\ \hat{c} \in \{c, \bar{c}\}$, for student t, by following the clapping function in Eq. 7 in [232]. Then, the equation to compute the embedding becomes:

$$\mathbf{v}^l_{t_{\hat{c}}} = \mathcal{Z}^l_{t_{\hat{c}}} \times_m \left\{\mathbf{U}^{(m)}\right\}^l_{\hat{c}}, \forall\ \hat{c} \in \{c, \bar{c}\}\,, \tag{4.20}$$

which follows a typical computation of the core tensor, given the data tensor and its factor matrices.

It is worth noting that since PARAFAC decomposition does not enforce the orthogonal property in each of the factor matrices, direct computation of Eq. 4.20 is not feasible. Recent work by [62] proposes an efficient estimation of the core tensor for PARAFAC decomposition. By following the algorithm 1 in [62] to efficiently estimate $\mathbf{v}^l_{c_t}$ and then the embedding of student s at l-th level $\mathbf{v}^l_t \in \mathbb{R}^{2R_l}$ can be obtained by:

$$\mathbf{v}^l_t = \left[\mathbf{v}^l_{t_c} | \mathbf{v}^l_{t_{\bar{c}}}\right]. \tag{4.21}$$

Table 7: Dataset and Tensor Modes Description used in *iDisc*.

Dataset	Course	#Users	#Activity	#Areas	#Education	Stay	
edX	Course A	57,715	-	34	5	5	
	Course B	66,731	-				
	Course C	169,621	-				
				#Days	#Events	#Source	
XueTangX	Course 1	12,004	652,701	14	7	2	
	Course 2	10,321	877,805				
	Course 3	9,382	907,118				
				#Problems	#KC	#Views	#Duration
ASSISTments	Year 2004	912	580,785	376	58	7	10
	Year 2005	2,392	521,751	266	59	4	10
	Year 2006	2,584	686,868	409	69	4	10

4.5 EXPERIMENTS

In this section, we conduct systematic experiments to evaluate the quality of *iDisc*. In following, we first describe the data used for the experiments. The content of the rest of this section is structured to answer the following questions:

- Can *iDisc* reveal meaningful patterns?
- How does *iDisc* perform in comparison with state-of-art methods from predictive modeling in learning analytics?
- Can *iDisc* scale for the dataset at a massive scale?

4.5.1 Data

We experiment with nine courses/sessions from three publicly-available MOOC platforms: edX, ASSISTments, and XueTangX. The statistics of the datasets are provided in Table 7.

edX. The edX [173] dataset is comprised of de-identified data from "Introduction to Computer Science" (Fall 2012, Spring 2013, and Summer 2013) from MITx (Course A and B in Table 7) and HarvardX (Course C). This dataset does not have detailed event logs. However, the data are aggregated records, where each record represents the summary statistics for one individual's activity in the edX course with her demographic information. We select the three most popular courses from this dataset. For this dataset, we construct a 34x5x5 tensor as *Area*×*Education*×*Stay*. Our goal is to predict whether the course completion certificate is earned by a student at the end of the course.

XueTangX. XueTangX [252] is one of the largest MOOC platforms in China. The full dataset includes 79,186 students enrolled in 39 classes. Each enrollment is associated with a log of the students activities, including watching lecture videos, working on course problems, accessing course modules, and so on. In total, there are 8,157,277 activity logs, and the longest lifetime of enrollment is five weeks. We take the three most popular courses from this dataset. The dataset statistics are shown in Table 7. We use the first two weeks to learn the factor matrices by constructing a 14x7x2 tensor as *Day*×*EventType*×*EventSource*. The goal is to correctly predict course completion.

ASSISTments. ASSISTments [83] is an online tutoring system used by more than 50,000 students around the world [49]. On ASSITments, students attempt to solve problems and receive feedback on those attempts. To assist the learning process, each problem is also associated with multiple knowledge components. We take the public dataset of the Math course on ASSISTments over three years (2004, 2005, and 2006) and the dataset characteristics are shown in Table 7. For each dataset, we construct a four-way tensor as *Problem*×*KnowledgeComponent*×*ProblemView* ×*ActionDuration*. Our aim is to classify the students as over-performing students and under-performing students, in terms of error-rate [1].

4.5.2 Qualitative Examination of the Patterns

In this section, we use "Introduction to Computer Science" on MITx during Spring 2013 to illustrate the outputs of *iDisc*. We first qualitatively examine the patterns, as well as the

[1] https://pslcdatashop.web.cmu.edu/help?page=terms#error_rate

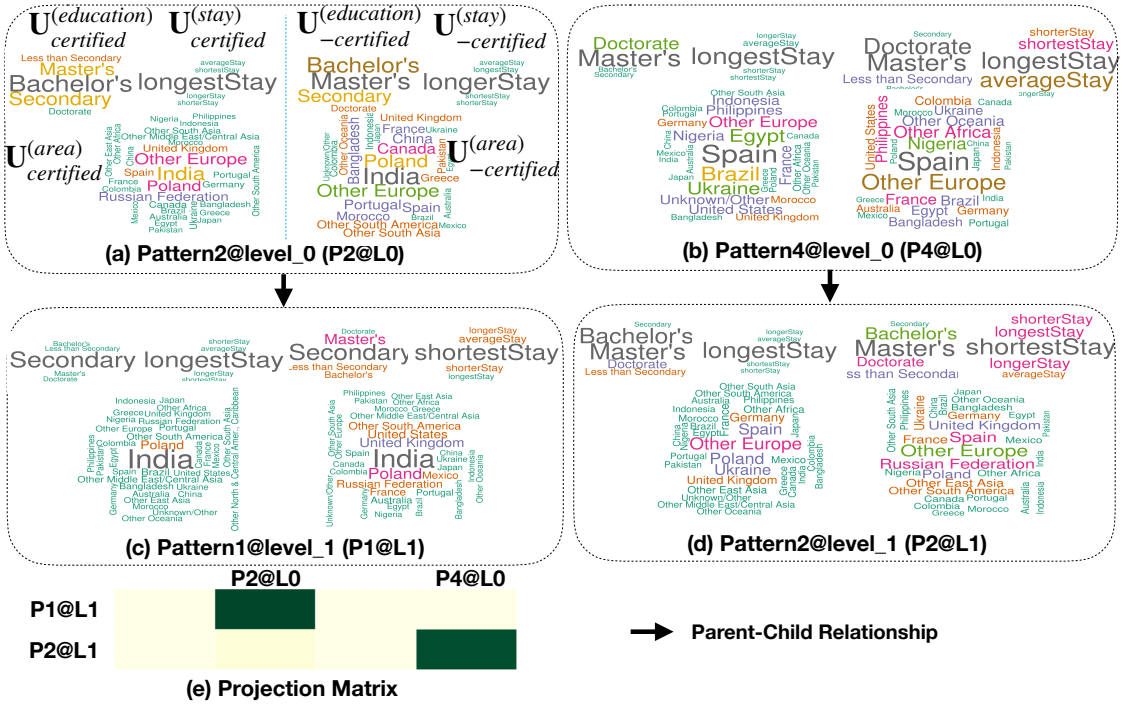

Figure 24: Model Output Illustration of *iDisc* from the MITx dataset. (e) shows the relationships between patterns in the first two levels; (a-d) show the associated patterns in (e).

pattern hierarchy generated and then explain the relationship between the patterns and the performance outcome.

4.5.2.1 Common and Discriminative Pattern Discovery Figure 24 describes the outputs by *iDisc*, with $\{R\}_l^L = \{4, 2\}$ (rank-4 at the first level and rank-2 at the second level). Particularly, Figure 24(e) shows the project matrix $\mathbf{P}^1 \in \mathbb{R}^{2\times 4}$ that represents the hierarchical relationships between the set of patterns at the first two levels, where darker colors indicate stronger associations. We observe that pattern #1 at level 1(P1@L1) is strongly associated with pattern #2 at level 0(P2@L0). This suggests that P1@L1 could be a child pattern for P2@L0. Similarly, we observe the parent-child relationship between P4@L0 and P2@L1. Since the projection matrix is shared by the factor matrices from the coupled tensors, it is

important to note that each of figures 24(a) through 24(d) refers to both patterns for the certified and non-certified group (i.e. as shown left and right in Figure 24(a)).

Due to the space limit, we only discuss the details of two sets of multi-way patterns as word clouds in Figure 24(a) for P2@L0 and Figure 24(c) for P1@L1. Figure 24(a) describes a subgroup of students that are from the most populated countries (e.g., the United States, India, and Poland, based on $\mathbf{U}^{(area)}$) with education background mostly from Secondary to Masters ($\mathbf{U}^{(education)}$), and most of whom tend to stay on the edX platform ($\mathbf{U}^{(stay)}$) for a relatively long period. While the pattern from the certified group of students shares almost identical distributions in the area and educational background, what makes them slightly different was primarily the time spent on the platform; certified students($\mathbf{U}^{(stay)}_{\text{certified}}$) spend relatively longer than un-certified students ($\mathbf{U}^{(stay)}_{\text{-certified}}$). The set of patterns in Figure 24(c) is the child patterns of the Figure 24(a). Compared to their counterpart in level 0, they primarily describe the students from Indian (although the area distribution from the un-certified group of students spans more countries ($\mathbf{U}^{(area)}_{\text{-certified}}$)) with more focus on the middle level of education background (e.g., Secondary($\mathbf{U}^{(education)}_{\text{certified}}$)). The difference in their length of stay on the platform was more prominent in this set of patterns, where certified students have the longest stays with the edX platform ($\mathbf{U}^{(stay)}_{\text{certified}}$), and un-certified students generally have the least ($\mathbf{U}^{(stay)}_{\text{-certified}}$).

4.5.2.2 Simpson's Paradox Revisited

We performed multivariate logistic regression analysis to identify the patterns that can explain students' variation in obtaining the certificate for each level (M1 for level 0 and M2 for level 1). The dependent variable is whether or not the users are certified at the end of the course. The explanatory variables include the students' embeddings for the aforementioned patterns in Figure 24(a-d) in each level (e.g., $\mathbf{v}^1_{\text{certified}}(1)$ refers to the embeddings corresponding to P1@L1$_{\text{certified}}$). The embeddings are standardized to facilitate comparison among different variables.

Table 8 shows the estimated coefficients for M1 and M2. The only significant variable in M1 is the embeddings $\mathbf{v}^0_{\text{-certified}}(2)$ ($\beta = -0.113$, $p < .05$). This suggests that users who have shown more activities in line with pattern P2@L0$_{\text{-certified}}$ appear to have less chance to earn the certificate. M2 shows that both embeddings $\mathbf{v}^1_{\text{certified}}(2)$ ($\beta = 2.285$, $p < .01$)

Table 8: Explanatory variables are the students' embeddings that correspond to the patterns presented in Figure 24 (a-d) (e.g., $\mathbf{v}^0_{certified}(2)$ refers to the values in students' embedding vector $\mathbf{v}$ for P2@L0 from the certified group). *Note:* **:p<.05; ***:p<.01.

	Certified (M1)		Certified (M2)
$\mathbf{v}^0_{\text{certified}}(2)$	-0.058 (0.046)	$\mathbf{v}^1_{\text{certified}}(1)$	-0.282 (0.188)
$\mathbf{v}^0_{\text{certified}}(4)$	-0.004 (0.046)	$\mathbf{v}^1_{\text{certified}}(2)$	2.285^{***} (0.649)
$\mathbf{v}^0_{\text{-certified}}(2)$	-0.113^{**} (0.048)	$\mathbf{v}^1_{\text{-certified}}(1)$	-0.620^{***} (0.080)
$\mathbf{v}^0_{\text{-certified}}(4)$	0.026 (0.046)	$\mathbf{v}^1_{\text{-certified}}(2)$	0.733^{***} (0.061)

and embeddings $\mathbf{v}^1_{\text{-certified}}(2)$ ($\beta = 0.733$, $p < .01$) reveal a significant and positive effect towards earning the certificate, with $\mathbf{v}^1_{\text{certified}}(2)$ having a much larger effect size. On the other hand, M2 also shows a significant and negative effect of having a larger value in $\mathbf{v}^1_{\text{-certified}}(1)$ ($\beta = -0.620$, $p < .01$).

We can consider multi-way patterns as principal components in PCA that bridge the original feature interactions in the high-dimensional space and the associated *loadings*. In this case, we would expect students from one class to have higher loading scores associated with patterns that are extracted from the same class. The regression result in M1 shows that the un-certified group of students does have larger loading scores. However, that is true only in $\mathbf{v}^0_{\text{-certified}}(2)$ for P2@L0 . This is not surprising, because in level 0 the two sets of patterns are in fact very similar to each other, as shown in Figure 24. The results in M2 confirm this expectation, with significantly higher loading scores $\mathbf{v}^1_{\text{certified}}(2)$ for certified students and significant higher loading scores $\mathbf{v}^1_{\text{-certified}}(1)$ for un-certified students. Contrary to our expectation, the certified group of students also have higher loading scores in $\mathbf{v}^1_{\text{-certified}}(2)$ (although much lower than $\mathbf{v}^1_{\text{certified}}(2)$). This could be explained by our observation in Figure 21, where P2@L1 captures a cluster of highly-educated students from certain European

Table 9: Classification Results in Accuracy in Different Courses in Comparison with Existing Methods.

Dataset		edX Course A	edX Course B	edX Course C	XueTangX Course 1	XueTangX Course 2	XueTangX Course 3	ASSISTments Year 2004	ASSISTments Year 2005	ASSISTments Year 2006
Baselines	Raw	90.50	91.55	87.34	61.99	69.15	69.37	64.23	57.20	60.36
	LDA	76.83	75.04	75.17	66.93	71.39	70.44	62.79	66.50	66.44
	FM	93.98	94.21	88.89	65.31	70.50	69.43	74.10	69.32	70.20
	LadFG	-	-	-	68.35	72.63	73.56	-	-	-
	SDCDNTF2	94.42	96.55	93.43	67.23	71.50	71.27	61.52	59.79	60.94
	SDCDNTF4	94.37	96.47	93.46	67.15	72.56	71.19	61.49	62.32	60.88
	PairFac	94.44	95.54	93.19	67.40	71.96	72.22	69.54	67.59	70.26
Proposed Method	*iDisc*-1st	94.10	95.28	93.01	66.56	71.24	72.36	72.82	69.47	66.50
	iDisc-2nd	**95.37**	**96.86**	**94.58**	**69.39**	73.27	73.13	**78.37**	**72.84**	**73.26**
	iDisc-Comb.	94.79	96.14	93.76	69.35	**74.00**	**74.12**	77.47	72.65	71.49

areas. They have a much higher chance of obtaining the certificate, regardless of having the longest stay or shortest stay with the platform.

Summary. We qualitatively examine the outputs generated by *iDisc*. The results show interesting properties of the proposed method. Our model reveals common and discriminative patterns at each level with their relationship explained via the projection matrix. Our regression analysis first explains the discriminative capability of the students' embeddings based on this set of patterns. More importantly, the analysis validates that the students' embeddings can be used to measure the relationship between the performance outcome with multi-way patterns from *iDisc*.

4.5.3 Quantitative Comparison

In this section, we report the results from the quantitative experiments in comparison with existing work commonly used in predictive analytics. Specifically, we conduct a classification task, in which the goal is to predict the students' performance at the end of the course defined in Section 5.1.

4.5.3.1 Baselines We include baselines that are commonly seen in the area of predictive analytics in educational data mining as:

- Raw. We use the raw activity counts each day as features to train classifiers for prediction. This is the most common approach in predictive modeling for MOOCs.
- LDA. Coleman et al. [41] use LDA to capture the temporal element of the behavior data. We first discover the latent behavior patterns from Raw features with a varying number of topics, and use the topic membership of each student for the classification task.
- LadFG [180]. As one of the most cited works in MOOC predictive modeling, LadFG is a latent dynamic factor graph model that finds a mapping from students time-varying attribute tensor to the observed learning outcome. We only evaluate the performance of LadFG in XueTangX dataset because it is the only one of the three that contains the necessary temporal dynamics.
- Factorization machines (FM) [184]. Factorization machines have been proposed and successfully applied to recommendation and prediction tasks. As the factorization model projects the input feature space into a latent space, it enables the learning of more complex interactions between features. We first convert each dimension as dummy variables for each student, and then concatenate all dimensions as a wide feature matrix.

It is worth noting the recent use of Deep Neural Nets (DNN) and their variants have shown promising performance compared to conventional machine learning approaches (e.g., [100, 227, 236]). However, the lack of interpretability of these models prevents their further application in problems driven by both interpretations and performance gains. We also compare *iDisc* with the existing work on discovering the common and discriminative patterns from multi-way data.

- `SDCDNTF`. `SDCDNTF` extends [101] and learns the common and discriminative patterns with different ranks. The input to the model consists of the rank and the number of shared patterns, along with the coupled tensors.
- `PairFac`. `PairFac` [232] learns the common and discriminative patterns with different ranks. Comparing to `SDCDNTF`, it does not require the input of the split.

We would like to point out that standard tensor factorization could serve as another baseline to compare with, in which students or their class reside as one of the dimensions and the corresponding factor matrix can naturally become features for downstream prediction tasks. However, we did not include it for two reasons: first, because the aforementioned baselines work as inductive models, where unseen students can be predicted based on the learned parameters, while simple tensor factorization serves as a transductive model and only predicts for the students that are available in the factorization; and second, because student populations on MOOC platforms can be of any size, from small to very large, the efficiency of standard tensor factorization with a large dimension size could be a practical problem for its real-world application.

4.5.3.2 Experiment Settings For each dataset, we draw a training set of students from each class with replacement, and then obtain the embeddings of the out-of-bootstrap students. For this set of students, we perform a five-fold cross-validation with a k Nearest Neighbors classifier. We conduct five independent trials of this experiment and report the average classification accuracy. We select accuracy since both the training and testing dataset are constructed in a way that each class has an equal amount of students. For `SDCDNTF`, we experiment with α, β and $\gamma \in \{10^{-5}, 10^{-4}, 10^{-3}\}$. Finally, we set α and β in the same range, and $R = 6$ for both `PairFac` and `SDCDNTF` and derive two versions of `SDCDNTF` using $K \in \{2, 4\}$. To make a fair comparison with `PairFac` and `SDCDNTF` , we use two-level pattern discovery with rank-4 and rank-2 in each level for *iDisc*, respectively. For LDA and FM, we experiment with a varying number of topics /factors and report the best performance. For LadFG, we keep the suggested parameters from their paper.

4.5.3.3 Experiment Results Table 9 shows the classification results. The raw features perform poorly, especially in XueTangX and ASSISTments dataset, with the score on accuracy in the range of 60-70%, which suggests the difficulty of the prediction task. LDA saw different performance, with noticeable drops in the edX dataset in comparison to raw features. We conjecture that the construction of the raw features results in a high dimensional and sparse feature space, which could potentially cause LDA to suffer from learning merely

meaningful latent topics. Factorization machines slightly improve the performance. FMs can be considered as a generalization of tensor factorization with the additional modeling of interactions within each dimension [184]. Although we observe noticeable gains from FMs over Raw features and LDA, FMs do not perform as well as tensor-based methods. We suspect there might be two reasons for this: 1) the current feature space might not be as well tuned for general prediction tasks as it is for the more commonly seen recommendation tasks; 2) compared to tensor-based methods that only consider the interactions between different dimensions, FMs could potentially over-fit the interaction effects between and within dimensions in the training data. We observe that LadFG achieves large gains over the raw features for the XueTangX dataset. This indicates there could exist some hidden patterns that can capture the temporal elements of the behavior data. We also notice that tensor-based models such as `PairFac`and `SDCDNTF`perform better than LDA, especially in Course 2 and Course 3. This suggests that by systematically considering the multi-way interactions, the performance could be further improved. Finally, the best performance of *iDisc* is statistically comparable with the state-of-art LadFG and significantly better than the rest of the baselines. While LadFG is geared towards student performance predictions, *iDisc* can provide comparable prediction performance as well as meaningful patterns.

Summary. *iDisc* constructs students' embeddings that integrate relations between the multi-way interactions and the performance outcome. The quantitative experiment demonstrates the discriminative capability of *iDisc*, and *iDisc* outperforms the baselines in nine datasets from three MOOC platforms. Higher-level embeddings from *iDisc* have shown stronger discriminative powers over ones from the lower levels, which we will discuss in next section. Since there is no trivial solution in determing the rank of the tensor decomposition, we experimented with different rank settings for *iDisc* (e.g., $\{2, 4\}$, $\{3, 3\}$) and this observation still holds.

4.5.4 Parameter Sensitivity Analysis

In this section, we conduct experiments to analyze the sensitivities of the parameters involved in *iDisc*. We first report our observations in selecting a number of levels and then discuss

the rest of parameters. Due to the space limit, we only detail our experiments with the ASSISTments year 2004 dataset.

4.5.4.1 Selection of Levels One key parameter for *iDisc* is the level that we choose for common and discriminative pattern discovery. Section 5.3 has shown the second level representations have more discriminative power than the ones at first level. As the common and discriminative pattern discovery at each level requires both residual tensors ($\mathcal{E}_c$ and $\mathcal{E}_{\bar{c}}$) and from which, the auxiliary tensors ($\mathcal{S}$ and $\mathcal{Z}$) are computed, we suspect that the sparsity of these tensors could influence the quality of the pattern discovery.

To verify this conjecture, we set to examine the relationship between the sparsity and the classification performance. We run *iDisc* on our dataset with ten levels of pattern discovery and set the rank of each level to be four for a fair comparison between the levels. Figure 25(a) shows that the sparsity of the tensors from 0 to the 9-th level. We observe a sharp decrease at the 1st level, and the sparsity continues to decrease gradually. Then, we compute *class separation ratio* (*csr*) as

$$csr = (\frac{dist_{betweengroup}}{dist_{withingroup}} - 1) \times 100\% \tag{4.22}$$

and use it as a proxy to measure the difficulty to separate the students belong to one class from ones to the other. In this case, the larger *csr* is, the better two classes are separated. Figure 25(b) shows the corresponding *csr* at each level. As the level increases, we first observe the increase in the class separation. However, this is followed by a decrease and *csr* eventually stabilizes. These two figures suggest that as the sparsity of the tensors involved in the subsequent levels increases, *iDisc* might tend to pick up signals that are not contributing to the class separation, as much as it could with less number of levels. As *iDisc* is an iterative approach, we suggest in the real-world applications, users can empirically choose the number of levels to the point that the accuracy cannot be further improved, should more levels are preferred by the domain experts.

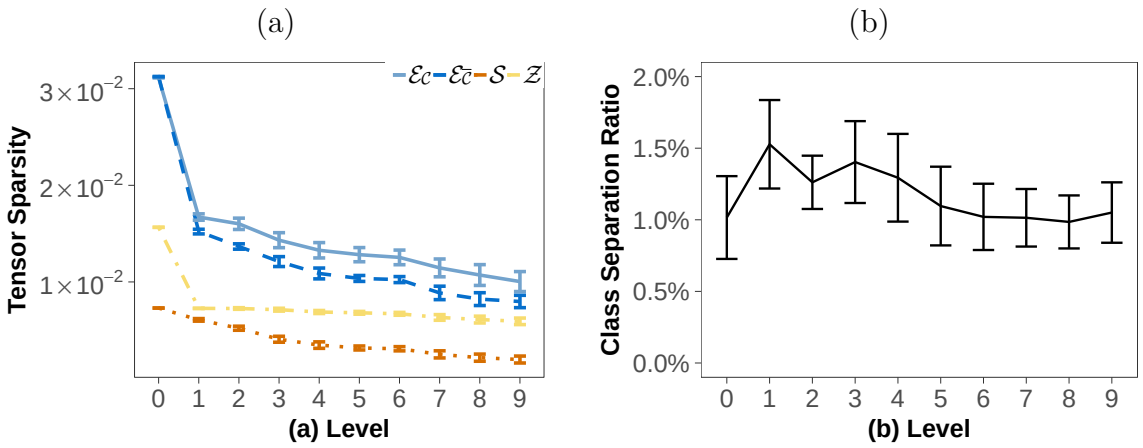

Figure 25: Level Sensitivity Analysis of *iDisc*. (a) shows the sparsity of the tensors involved in *iDisc* at different levels in ASSISTments dataset year 2004. (b) shows the corresponding class separation ratio based on different levels of patterns by *iDisc*.

4.5.4.2 Model Parameters This section aims to test the effectiveness against various settings of model parameters and to provide guidance on the parameter tuning. We follow the same experiment set-up as introduced in Section 5.1 with the focus on the set of parameters in $\{\lambda_0, \lambda_1, \lambda_2, \lambda_3\}$ in range $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}\}$. For each experiment, we change one of the parameters while keeping the rest fixed. Figure 26 shows the parameter sensitivity with respect to the classification accuracy. The performance is relatively more stable with respect to the change of λ_1 and λ_3. λ_1 enforces components aligned to become similar. As a larger λ_1 tends to more aggressively push the components to be alike, the drop in classification performance is rather expected. λ_3 specifies the weight on the l_1 norm of the projection matrix. With a larger λ_3, the accuracy first increases and then becomes stabilized. Since a larger λ_3 leads to a more sparse solution of the hierarchical relationships. We suggest the choice of λ_3 to be a trade-off between the accuracy and the hierarchical relationship among the patterns. λ_0 is used to control the weight on the factorization of the auxiliary tensors $\mathcal{S}$ and $\mathcal{Z}$. A small λ_0 is enough to push the factorization of the tensors towards discovering common and discriminative patterns. However, when λ_0 is too large, the learning

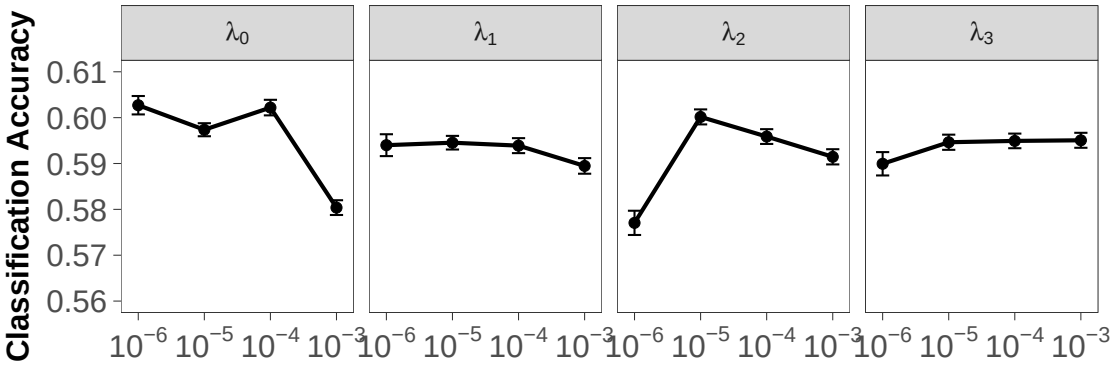

Figure 26: Parameters Sensitivity Analysis in Classification Task.

of these patterns can overshadow the decomposition of the coupled tensor and thus lead to the drop in performance. λ_2 controls the weight to learn the pattern hierarchy. We notice that classification increases dramatically when λ_2 raises from 10^{-6} to 10^{-3}. However, as λ_2 increases, the accuracy starts to drop. We conjecture that the hierarchical relationship between the patterns can aid the classification process. However, once the emphasis on the learning of hierarchy becomes too much, the hierarchy tends to be artificially boosted, and thus, the learned patterns are not able to capture much variance from the data.

4.5.5 Scalability

Since many of the education platforms have seen exponential growth in usage, scalable solutions of learning analytic are another critical aspect of adoption. In this section, we test the scalability of *iDisc*. In this experiment, we choose the ASSISTments dataset for the year of 2006, since it has the largest tensor settings in our experiment. We run *iDisc* with a varying number of entries in the data tensor, from $\{10^2, 10^3, 10^4, 10^5, 10^6\}$. Table 10 reports the average running time per epoch, and we observe that the running time scales almost linearly with the exponential increase in observations in the tensor. This result is consistent with the analysis in Section 4.1.4.

Table 10: Scalability Analysis For *iDisc* (Running Time For Varying Number of Observations in the Tensor.

#observations	10^2	10^3	10^4	10^5	10^6
running time/epoch	0.31s	0.35s	1.13s	1.23s	2.05s

4.6 SUMMARY

In this chapter, we present a tensor-based learning framework, *iDisc*, to perform common and discriminative pattern discovery at multiple levels for understanding of high-dimensional student behavior and performance prediction in MOOCs. We first use tensors to represent each user's behavior, and construct coupled tensors to aggregate behavior for users with contrasting performance groups. Then, we iteratively identify the shared and distinct behavioral patterns at various levels, while revealing the hierarchical relationship between them to further increase the interpretability of the output. Finally, we use these patterns as anchors to generate the students' latent representation for down-stream performance prediction. Our qualitative examination of the patterns has shown the multi-level, multi-aspect and hierarchical characteristics of behavior patterns on the edX platform. The quantitative experiments, compared to both traditional predictive methods as well as existing discriminative tensor factorization models, suggest promising results by *iDisc* in several datasets from different MOOC platforms.

To the best of our knowledge, this is the first attempt to tackle the joint problem of discriminant tensor factorization and hierarchical pattern discovery for understanding such behavior on MOOC platforms. This enables the in-depth comprehension of students' multi-way behavior dynamics, as well as its association with course performance. Nevertheless, one of the limitations is that it merely provides the relationships between the latent multi-way interaction and the performance outcome, with no intention to draw causal reasoning between them. In practice, *iDisc* can be developed as a plugin for MOOC platforms, where

instructors can examine the multi-aspect contrasting behavior and connect the difference to the course outcome. Considering the XueTangX platform, one of the multi-aspect patterns could refer to a set of events at the beginning of the course that trigger from the server. if *iDisc* reveals its positive association to the success of students' course end performance, this pattern can be used as guidance of promotions for both the instructor and the platform to improve the students' learning outcome. Last, but not least, compared to other tensor factorization methods, *iDisc* provides a more efficient exploration of the multi-aspect patterns due to its multi-level nature. However, we understand that the interpretation of the multi-aspect pattern itself is not straightforward in general. In our future work, we would like to follow a more human-centric approach and develop a visual analytic system that helps domain experts interpret and understand the multi-aspect patterns.

5.0 FACIT: FACTORIZING TENSORS INTO INTERPRETABLE, SCRUTINIZABLE, AND FINE-TUNABLE PATTERNS

In this chapter, we wanted to provide a unified and generic visual analytic system that addresses interpretability in the process of **Multiplex Pattern Discovery**, **Multifaceted Pattern Evaluation**, and **Multipurpose Pattern Presentation** in a general unsupervised pattern mining setting from multi-aspect data. We introduce *FacIt* (Chapter 5), a generic visual analytic system that directly factorizes tensor-formatted data into a visual representation of patterns to facilitate result interpretation, scrutinization, information query, and model selection and refinement. The idea of multipurpose pattern presentation is threefold: *1) the results are presented in understandable terms that experts can efficiently explore (Chapter 5.5.3, 5.5.4, and 5.5.6); 2) human information needs are learned through experts' interactions with patterns and further incorporated it into the factorization process (Chapter 5.4.2); 3) the novel presentation of the results empowers experts with a more rigorous evaluation schema, including model quality statistics (Chapter 5.5.2), qualitative pattern validation (Chapter 5.5.4), qualitative pattern utility (Chapter 5.5.5).*

5.1 INTRODUCTION

As a dimension reduction technique for high-dimensional datasets, tensor factorization has been widely used to identify latent patterns from multi-aspect data. Similar to other dimension reduction methods, such as singular value decomposition (SVD) [69] or latent dirichlet allocation (LDA) [22], tensor factorization helps users extract latent patterns with the noise of raw data removed. Such patterns tend to be more abstract and compressed, and generally

better describe correlations and interactions within the original set of dimensions. In fact, tensor factorization has been used to discover multi-aspect patterns that jointly describe underlying data phenomena, in many real-world applications, such as social network analysis [121], web search [10], brain data analysis [166], and healthcare [87,144]. Despite its wide range of applications, identifying insightful patterns from Tensor Factorization still poses three challenges:

- (1) Mismatch between human information need/interest and optimization goals: Tensor Factorization is optimized to minimize discrepancies between data and model. However, this goal often does not satisfy human information need. For example, users of Tensor Factorization might sacrifice its fit for a sparse representation that is easier to interpret.
- (2) Mismatch between experts' domain knowledge and data-driven models: In real-world applications, data can be noisy and therefore, a data-driven model with an adequate fit does not translate to one that experts can use their domain knowledge to interpret.
- (3) Mismatch between factorization results and human understandability: Factorization results often do not readily translate to how humans see things as clustered or close to one another.

While these challenges call for more user-driven pattern discovery methods from multi-aspect data, there has been little work to understand the specifics of human information need in the process of Tensor Factorization. Viola [26] and TPFlow [125] are among the few attempts to understand users' need. However, their primary focus is on applications within a spatio-temporal context and rather than more general situations.

To address the above challenges, we present *FacIt* (pronounced as *facet*), a generic visual analytic system that factorizes tensor-formatted data into a set of visual representations of patterns to facilitate model selection and refinement, result interpretation, and pattern scrutinization. Specifically, our work makes the following key contributions:

- **Task Analysis and System Design**: We conduct interviews with users of Tensor Factorization from three different domains to understand the information need based on their experience of tensor-based analysis. From this, we formalize a set of analytical

tasks and requirements applicable to generic tensor-based analysis. We propose a system design that closely follows design requirements distilled from the task analysis;

- **Algorithm**: We develop a novel weakly supervised tensor factorization model that leverages users' feedback to iteratively fine-tune tensor factorization outputs;
- **Visualization and Manipulation**: We propose a suite of visualization design tools that support effective model evaluation, iterative pattern fine-tuning, and efficient pattern scrutinization.

We demonstrate the power of the *FacIt*with three usage scenarios across different domains. Experts' feedback from in-depth interviews confirms its effectiveness, usefulness, and general applicability.

5.2 REQUIREMENT ANALYSIS

In this section, we first present tensor preliminaries on tensor. We then introduce design goals and analytical tasks to define the requirements of our system functionality. We provide examples to make requirements concrete.

5.2.1 Tensor Preliminaries

Although the tensor preliminaries are described in Chapter 2.1, we present the essential background in a more concrete context using the example of NBA shot dataset.

Tensors. A tensor $\mathcal{X}$ is a multidimensional array which is an extension of a scalar x, a vector $\mathbf{x}$, or a matrix $\mathbf{X}$ to a higher dimension. When a tensor has n-dimensions or *modes*, it is called n-way tensor, where each way's dimensionality is determined by the number of *items*. For example, a three-way tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ has three modes with dimensionalities I_1, I_2, and I_3, respectively. A non-negative tensor $\mathcal{X} \in \mathbb{R}_{+}^{I_1 \times \ldots \times I_M}$ is a tensor where all entries are non-negative values, which commonly applies to situations where data represent numbers of observed instances or *counts*. Fig. 1 shows one example of a tensor with three dimensions from NBA shot data in the 2014-2015 season [171]. Each entry x in the tensor represents

the number of shots taken by a given player at a given zone in the court in a given quarter.

Tensor decompositions can be considered as higher-order generalization of the matrix singular value decomposition (SVD) and principal component analysis (PCA). The CANDECOMP/PARAFAC (CP) [27, 80] and Tucker decomposition [110, 221] are the two most popular tensor decomposition approaches. We focus on a CP model in this work, but the proposed system and visualization design is extensible to a Tucker model. A *CP decomposition* of an M-way tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_M}$ finds a set of *factor matrices*, $\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(M)}$, that approximate the tensor as the sum of R vector outer products. It can be concisely expressed as: $\mathcal{X} \approx [\![\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(M)}]\!] \equiv \sum_{r=1}^{R} \lambda_r \mathbf{u}_r^{(1)} \circ \ldots \circ \mathbf{u}_r^{(M)}$, where the m-th factor matrix $\mathbf{U}^m = \left[\mathbf{u}_1^{(m)} \cdots \mathbf{u}_R^{(m)}\right]$ is the combination of the vectors from the R components. $\lambda_r \in \mathbb{R}^R$ is often used to absorb the respective weights during normalization of the factor matrices columns and $\circ$ represents outer products. Fig. 1 shows an example process of CP decomposition which factorizes the *player* $\times$ *zone* $\times$ *time* tensor from NBA shot data (explained in section 3.2) into a set of components.

Items, Descriptors and Patterns. We refer to each entry i for $i = 1, \ldots, I_m$, as an *item* of the m-th dimension in the tensor. In Fig. 1, in the player dimension, each item refers to a player in the set of players, e.g., {Stephen Curry, Lebron James, Klay Thompson, ...}. We denote the vector $\mathbf{u}_r^{(m)} \in \mathbb{R}^{I_m}$ as a *descriptor* consisting of entries $\langle \mathbf{u}_{ir}^{(m)} \rangle$ for $i = 1, \ldots, I_m$ from the m-th dimension that describes the contribution of the i-th *item* i to the r-th component. For a non-negative tensor, it is often useful to constrain the descriptor to take non-negative values to facilitate the interpretability of occurrence-likelihood, i.e., $\mathbf{u}_r^{(m)} \in \mathbb{R}_+^{I_m}$ where an entry value $\mathbf{u}_{ir}^{(m)}$ can be considered as how likely the i-th item is associated with the r-th component. The r-th component or *pattern* is a collection of vectors from each mode $C_r = \{\mathbf{u}_r^{(1)}, \ldots, \mathbf{u}_r^{(M)}\}$. In Fig. 1, each pattern of shot behaviors consists of three descriptors: player, zone, and time. In this work, "pattern" and "component" are used interchangeably while "pattern" also refers to a component as a visual representation.

Rank. In the aforementioned tensor decomposition, R denotes the specified *rank* – the number of components. In practice, the rank is determined numerically by fitting various rank-R models for $R = 1, 2, \ldots$ until a "good" model is found. However, we argue that the numerical "goodness" of fit should not be the only criterion for specifying the rank.

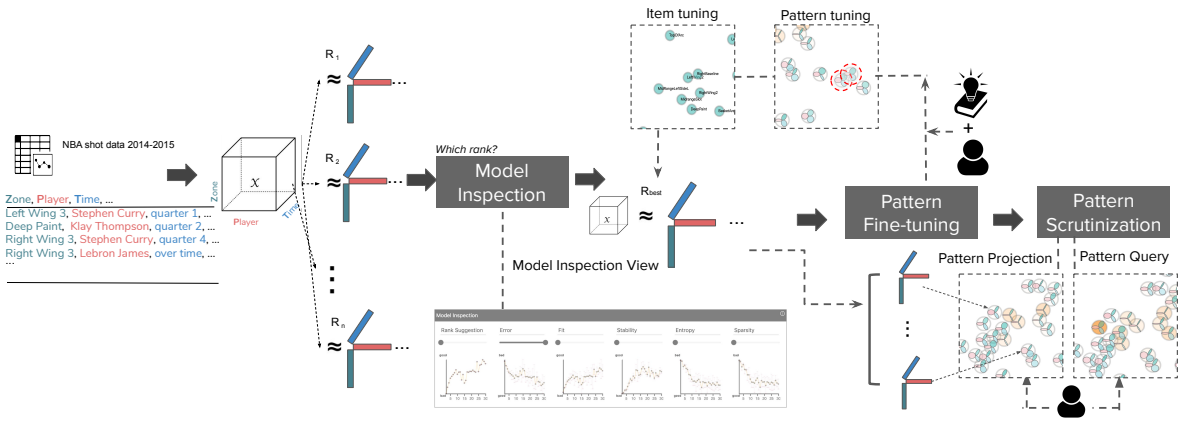

Figure 27: **System Overview**. *FacIt* consists of three key modules: Model Inspection to assist users to diagnose for a proper rank; Pattern Fine-tuning users a human-in-the-loop to iteratively update the model outputs; Pattern Scrutinization provides various mechanisms for efficient pattern exploration.

Other criteria such as model compactness and interpretability are also important for a tensor decomposition results to be practically useful. In this work, we tackle the problem of the selection of rank as a part of our task.

5.2.2 Procedure and Data

The design of *FacIt* follows the nested model described in [149], which is an iterative analytic process with one expert from each of three different application domains – sports analytics, online purchases, and public policy analysis. Despite the diverse application domains, our experts all have extensive experience in computational data analytics in their respective domain. Our domain experts are further selected based on two criteria: 1) they have knowledge of tensor factorization; 2) they are comfortable using an off-the-shelf package to run tensor factorization and have used it in their analytic tasks in the past. We use three datasets from these applications as motivating examples and use the last one to evaluate our system. The first application is the analysis of NBA shots data in season 2014-2015 [167]. The second is the analysis of online coupon purchases [93]. The third is the analysis of policy adoption in

Table 11: Dataset and Tensor Modes Description in *FacIt*.

Domain	Dataset	Tensor Setting	Tensor Entry	Size
Sports	NBA shot data (2014-15)	Period, Player, Zone	#shots	$[5, 15, 14]$
Business	Ponpare Coupon Purchase	Genre,Sex,Age, Price, Period	#coupon	$[13, 2, 13, 10, 7]$
Politics	Policy Adoption in U.S.	Subjct, Year, State, Keyword	#adoption	$[16, 26, 50, 18]$

the United States. Table 11 summarizes the three datasets.

Over the course of six months, we held weekly meetings with each of the experts. During our early meetings, we discussed system design requirements in experts' respective domains. Then, we proposed system prototypes, and demonstrated them to the experts to gather their feedback. Improvements were made iteratively throughout this process.

5.2.3 Design Goals

Based on a thorough literature review and interviews with the experts, we identify the following design goals for visual analytic systems to assist in tensor-based analysis:

G1. Effectiveness: How can we assist users in evaluating the effectiveness of model configuration? The result of tensor factorization is highly dependent on the configuration of the *rank*. However, there is no trivial algorithm to determine the optimal rank. Our experts suggest that the system should help the user decide which rank leads to the most effective decomposition by providing a set of quality measurements and letting users inspect the patterns associated with a specific rank.

G2. Efficiency: How can we assist users in exploring patterns more efficiently? Our experts mentioned that the process of exploring patterns is an iterative and time-consuming process. It takes a great amount of time to examine all the patterns until discovering the meaningful ones. In the regard, the system should present a high-level pattern summary which allows users to instantly locate patterns of interests.

G3. Interpretability: How can we better understand a pattern? As mentioned by our experts, it is a time-consuming task to interpret patterns by examining multiple charts

of descriptors. To complement existing approaches, which typically present descriptors side-by-side [26, 125, 193], the system should provide a space-efficient and well-balanced visual representation to explain multiple aspects of a pattern.

G4. Comparison: How can we compare patterns? While previous studies have addressed the issue of comparing multiple patterns (e.g., [5, 56, 57, 168, 193, 204, 211, 233]), experts were interested in selecting a pattern that matched their interests, and identifying other similar patterns in terms of a certain dimension or a combination of several dimensions. Therefore, the system should intuitively represent patterns based on how they are similar to each other.

G5. Human-In-The-Loop: How can we leverage experts' domain knowledge to refine the patterns? While a variety of tensor factorization variants have incorporated domain-specific knowledge (e.g., [4, 6, 9, 11, 92, 121]), most of them require such knowledge to be incorporated before the analysis. Further, most approaches produce factorization results in a static manner – that is, analysts have very little or no way to incorporate their domain knowledge flexibly other than simply changing the hyper-parameters (e.g., rank or the random initialization). Therefore, the system should allow users to interact with the tensor model. Users should be able to provide feedback that steers the factorization interactively.

5.2.4 Analytical Tasks

To meet the design goals mentioned above, we summarize the analytical tasks as follows:

T.1: Present comprehensive model statistics. The system should provide a comprehensive set of essential metrics (T.1.1) associated with the quality of the tensor factorization. The system should allow users to configure the rank settings based on their preferences and facilitate an understanding of the trade-offs (T.1.2). The system should then immediately present the patterns as an output of the selected rank to enable a quick quality check (T.1.3). (**G1**)

T.2: Present an overview of patterns. The system should provide users with an overview of patterns to illustrate the relationships between the patterns (T.2.1). Users should

be able to explore patterns which have varying degrees of contribution to the model (T.2.2). (**G2**).

T.3: Multi-facet pattern query. The system should provide an interaction mechanism that allows users to query patterns that match their items of interest (T.3.1). Upon issuing of queries, the system should present a ranked list of patterns that are most relevant to the query (T.3.2). Since a query may consist of a combination of multiple items, the system should include a feature that allows users to keep track of querying history (T.3.3). (**G2**)

T.4: Visualize the multi-aspect characteristics of patterns at multiple scales. We need to design a set of visualizations for patterns that preserve and integrate their multi-aspect nature at multiple scales. A high-level presentation should allow the users to effectively grasp the overall distribution of patterns and their summary characteristics (T.4.1). A low-level presentation should display on-demand details of the patterns with both quantitative distributions (T.4.2) and qualitative narratives (T.4.3). (**G3**)

T.5: Encode the multi-scale comparison between patterns. The system should first summarize similarities and differences between descriptors across patterns (T.5.1). Moreover, the system should highlight the similar and discriminative items between patterns on demand so that users can immediately spot how two patterns differ from and concur with each other in each descriptor (T.5.2). (**G4**)

T.6: Support pattern fine-tuning based on users' feedback. The system should provide a set of visualization and manipulation tools for users to view, tune, and update model results. Particularly, the system should allow users to delete (T.6.1)/merge patterns (T.6.2), and also enable users to update items' position in the item space (T.6.3). (**G5**)

5.3 SYSTEM OVERVIEW

FacIt was designed to meet the requirements of domain experts in understanding and interpreting patterns from multi-aspect, real-world datasets. The system employs intuitive visualization designs and weakly semi-supervised tensor factorization techniques to help experts efficiently explore and interact with patterns from multi-aspect data. Fig. 27 illustrates

the major components of the system, which supports interactive pattern refinement and exploration mechanisms. The system has three key modules: (1) model inspection, (2) pattern fine-tuning, and (3) pattern scrutinization.

Model Inspection resolves mismatches between human information needs and optimization goals. Model inspection module is developed to provide a set of quality metrics for users to view the trend of objective measures with respect to the rank. *Iterative Pattern Fine-tuning* addresses discrepancies between users' domain knowledge and data-driven models. It features a novel weakly semi-supervised tensor factorization model that incorporates experts' feedback into the next iteration of pattern generation. By using a set of visualization designs and manipulation mechanisms, users have various ways of providing feedback to iteratively update the model based on their domain knowledge. *Pattern Scrutinization* addresses the mismatch between the factorization results and human understandability. This module translates factorization results into a set of artifacts that assist users when exploring the results. By using a novel glyph to encode each pattern, the system provides multiple efficient views of patterns to expedite the exploration process.

As shown in Fig. 27, the system takes a multi-aspect dataset in a tensor format (e.g., NBA shot data in the 2014-2015 season) and computes a set of quality measurements for each different rank setting. With a comprehensive understanding of the model performance from different perspectives, users can then select several plausible ranks (**G1**). The initial model outputs are a faithful reflection of the underlying data, which can be noisy and untrustworthy. Experts may have certain expectations of the model outputs based on their domain-specific experience. In the case of NBA shot data, experts can directly refine the patterns and manipulate item relationships to match their domain knowledge (e.g., in the player embedding space). This feedback is incorporated as experts' supervision in the tensor factorization process (**G5**). Experts can then use querying and visualization tools to explore the patterns (**G2, 3**), e.g., starting with a high-level pattern overview, querying for patterns with specific interests, and displaying pattern details on demand. Another way of understanding the patterns is through comparison. The system facilitates the comparison of patterns at multiple scales and perspectives (**G4**).

5.4 WEAKLY SEMI-SUPERVISED TENSOR FACTORIZATION

In this section, we introduce a novel weakly semi-supervised pattern discovery method based on tensor factorization. Given a set of multi-aspect data, the objective of this algorithm is to discover a set of patterns that are faithful to the data, but also reflect experts' domain knowledge of the dataset. We first present the users with patterns generated from a standard tensor factorization toolkit. Users then provide their feedback on the outputs through various interactions with the system. The system incorporates the feedback into the model and updates the patterns to match users' expectations.

5.4.1 Standard Tensor Factorization

Given multi-way data represented as a tensor, standard tensor factorization can be applied to extract an initial set of patterns. We use non-negative CP decomposition to factorize the tensor into a set of components. With a specified rank R, conventional tensor factorization seeks a set of latent factor matrices from a multi-way tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times\ I_2 \cdots\ \times\ I_M}$. The objective is to minimize the following cost function:

$$\mathcal{L}_0 = \left\| \mathcal{X} - \left[\mathbb{U}\right] \right\|^2, \tag{5.1}$$

where $\left[\mathbb{U}\right] = \left[\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(M)}\right]$ and $\mathbf{U}^{(m)}$ is the factor matrix that corresponds to the m-th dimension of tensor $\mathcal{X}$.

5.4.2 Weakly Supervised Tensor Factorization

In this work, we propose a weakly supervised tensor factorization algorithm that interactively allows the users to incorporate their domain knowledge and drive the factorization process. The model provides two kinds of feedback to aid the factorization process: 1) feedback on the patterns and 2) feedback on the items.

5.4.2.1 Feedback On Patterns Presented with a set of initial patterns from standard tensor factorization, our domain experts express their concerns with two phenomena. First, there are uninformative patterns that have little interpretable use. For example, one pattern could be almost uniformly distributed in all of its descriptors. They would like the system to remove such patterns. The other concern is that experts might see several patterns that are almost identical, which diminishes the values of interpreting these patterns individually. We seek to support interactions that allow the users to interactively delete or merge patterns (see Section 5.5.5 for details about interaction support). Given a collection of R components $\mathbf{C} = \{C_1, C_2, \ldots, C_R\}$ that a model outputs, each operation of merge or delete would reduce the rank of the model by one, i.e, $R-1$. Specifically, deleting the r-th pattern C_r is equivalent to the operation of removing r-th column $\mathbf{u}_r^{(m)}$ from $\mathbf{U}^{(m)}$, $\forall m \in \{1, \ldots, M\}$. Consider two patterns C_i and C_j to be merged. We first remove both from $\mathbf{C}$, and then add component $C_k = \{\mathbf{u}_k^{(1)}, \ldots, \mathbf{u}_k^{(M)}\}$ to $\mathbf{C}$, where $\mathbf{u}_k^{(m)} = \mathbf{u}_i^{(m)} + \mathbf{u}_j^{(m)}, \forall m \in \{1, \ldots, M\}$. After users' interactions, we seek to obtain a new factor matrix $\mathbf{U}^{(m')}$ for each factor matrix $\mathbf{U}^{(m)}$ as the reference matrix for next iteration of pattern discovery with the following regularization:

$$\mathcal{L}_1 = \sum_{m}^{M} \left\| \mathbf{U}^{(m)} - \mathbf{U}^{(m')} \right\|_F^2, \tag{5.2}$$

which forces the factorization outputs to be close to the reference factor matrix.

5.4.2.2 Feedback On Items Tensor factorization has been used as a way to discover latent relationships among items. A factor matrix $\mathbf{U}^{(m)} \in \mathbb{R}^{I_m \times R}$ can be considered as the item embeddings. With this information, item relationships can be further explored and used as a tool to verify the model correctness. Consider that in the latent space, if the items are clustered in a way that is not intuitive to domain experts, the data-driven factorization process is potentially flawed and will need correction. To this end, we design a set of interactions that allow users to adjust item relationships in a 2-D space based on their domain knowledge. We use such feedback as a reference for next iteration of the factorization. Specifically, we can infer a matrix $\mathbf{P}^{(m)} \in \mathbb{R}^{I_m \times 2}$ which captures the updated two-dimensional coordinates of the items in the m-th mode in the newly updated item space.

Given $\mathbf{P}^{(m')}$, we infer a pairwise distance matrix between the items based on heat kernel with a local scaling schema [248]:

$$\mathbf{W}_{ij}^{(m')} = \exp\left(- \frac{1}{\sigma_i \sigma_j} \left\| \mathbf{P}_i^{(m')} - \mathbf{P}_j^{(m')} \right\|^2 \right), \tag{5.3}$$

where $\mathbf{P}_i^{(m')}$ and $\mathbf{P}_j^{(m')}$ are the coordinates for item i and item j in the m-th mode, and σ varies for each item. σ_i is determined based on the distance between item i and its K-th nearest neighbor, where $K = \min(7, I_m)$. 7 is used because it has been shown to give good results [248]. With $\mathbf{W}^{(m)}$, we derive the following cost function:

$$\mathcal{L}_2 = \sum_{m}^{M} Tr(\mathbf{U}^{(m)^T} \mathbf{L}^{(m)} \mathbf{U}^{(m)}), \tag{5.4}$$

where $\mathbf{L}^{(m)}$ is the Graph Laplacian matrix, computed as

$$\mathbf{L}^{(m)} = \mathbf{D}^{(m)} - \mathbf{W}^{(m)}, \tag{5.5}$$

where $\mathbf{D}^{(m)}$ is a diagonal matrix whose entries $\mathbf{D}_{ii}^{(m)} = \sum_j \mathbf{W}_{ij}^{(m)}$.

We want to note that we can provide the interactions that distill experts' knowledge of the pair-wise relationship between patterns as another regularization to enforce the new set of patterns to exactly ensemble such relationship. However, our experts believe this process is not as straightforward as it is for the items because patterns can be a noisy reflection of the data and to adjust the pair-wise relationships between patterns, one needs to completely understand the entire pattern space.

5.4.2.3 Overall Objective Function To summarize, using the initial set of patterns from a standard tensor factorization, users can provide their feedback to the model by performing operations on the patterns and items. The system incorporates their knowledge and uses it as a regularization for the factorization in the next iteration. The overall objective function for factorization with supervision is as follows:

$$\mathcal{L} = \mathcal{L}_0 + \alpha \mathcal{L}_1 + \beta \mathcal{L}_2, \tag{5.6}$$

where α and β control the weights of their respective regularizations.

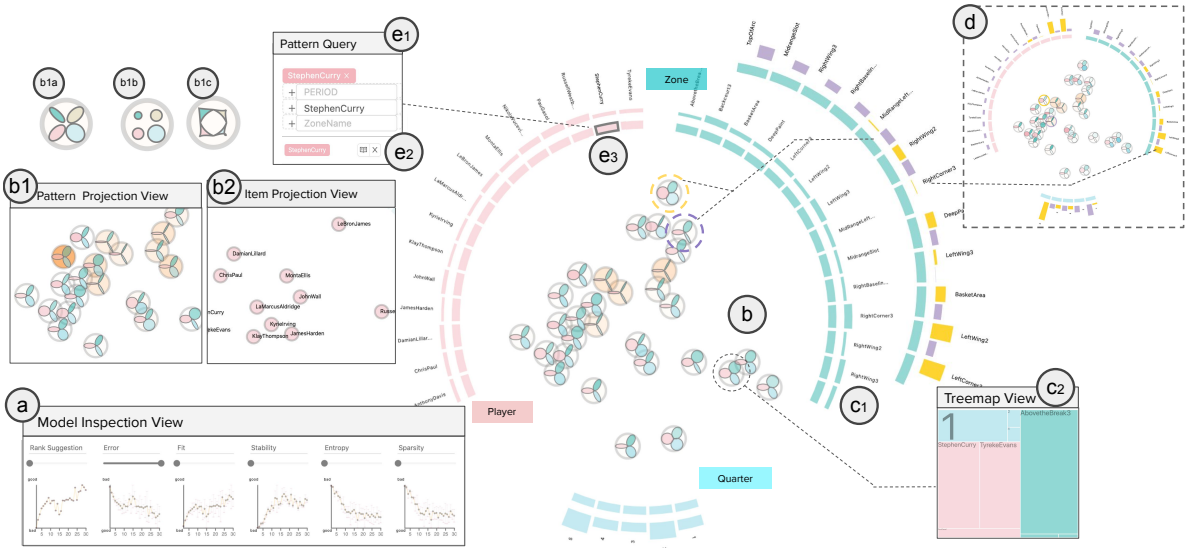

Figure 28: Using *FacIt*to interpret, fine-tune and scrutinize patterns based on tensor factorization from NBA shot data: (a) Model Inspection View provides various metrics of model sensitivity for selecting a desirable setting of rank from different aspects. (b) Pattern Projection View provides users high-lever overview of the entire pattern space. (c) Circular Bar Charts (c1) and Treemap view (c2) allow for examining the detailed content of patterns. (d) Pattern Comparison Mode allows users to analyze pairs of common and discriminative patterns and their associated items. (e) Pattern Query Mode enables users to retrieve most relevant patterns (e2) by query (text) input (e1) and item bars (e3).

5.4.3 Summary

FacIt features a weakly semi-supervised factorization model to iteratively incorporate domain experts' feedback. We wanted to note that *FacIt* shares this goal with Utopian [35]. However it is different in the tasks it addresses and therefore the optimization objective it follows. Utopian was proposed as an interactive topic discovery tool and limited to 2-dimensions, while our method is more generic for discovering, presenting and interpreting patterns from high-dimensional datasets.

Since the nature of the task is different, this leads to a different understanding of the

exact set of requirements from domains of literature and requires experts who are experienced in pattern discovery from high-dimensional data. As a result, the design requirements are not entirely the same. For example, pattern (topic) deleting/merging is one of the shared operations because, in both cases, users' control over patterns needs to be acknowledged. However, the item modification presented in this work does not pertain to the same purpose and therefore has different underlying mechanisms, compared to word modification in [35]. Unlike topic modeling, where a topic can be easily modified by changing the weights of its keywords, the complexity of modifying descriptor distributions changes dramatically with the increase of tensor modes. Indeed, domain experts did not appreciate this interaction when we were introducing this function to them. Instead of modifying the distribution of the items, experts preferred to manipulate items in the embedded space to incorporate their feedback. Through such straightforward interactions, the relationship between items becomes more aligned with experts' expectation.

Since objective function $\mathcal{L}$ is not convex with respect to $\left[\mathbb{U}\right]$, we aim to find a local minimum for $\mathcal{L}$ by iteratively updating each factor matrix in $\left[\mathbb{U}\right]$.

Let $\mathbf{U}$ represent the mode-m factor matrix. For simplicity of notation, we use $\overline{\mathbb{U}}$ to denote the set of factor matrices that correspond to modes other than m. Then, the optimization of $\mathbf{U}$ is equivalent to the following least squares loss functions:

$$\begin{aligned}\mathbf{U} \leftarrow \underset{\mathbf{U}\geq 0}{\operatorname{argmin}} \frac{1}{2}\Big(\frac{1}{n}\left\|\mathbf{X} - \mathbf{U}\ (\odot\overline{\mathbb{U}})^T\right\|_F^2\Big) \\ + \alpha\ Tr(\mathbf{U}^T\mathbf{L}\mathbf{U}) + \beta\left\|\mathbf{U} - \mathbf{U}^{'}\right\|_F^2,\end{aligned} \tag{5.7}$$

where $\mathbf{X}$ is the mode-d unfolding of tensor $\mathcal{X}$. Then the gradient update of $\mathbf{U}$ can be computed as:

$$\begin{aligned}\bigtriangledown_{\mathbf{U}}\mathcal{L} = \frac{1}{n}\left(\mathbf{U}(\odot\overline{\mathbb{U}})^T - \mathbf{X}\right)(\odot\overline{\mathbb{U}}) \\ + \alpha\mathbf{L}\mathbf{U} + \beta(\mathbf{U} - \mathbf{U}^{'}).\end{aligned} \tag{5.8}$$

5.5 VISUALIZATION AND INTERACTION

Based on the design requirements and analytical needs discussed above, we developed *FacIt*, an interactive visual analytic tool to facilitate the interpretation, revision, and scrutinization of tensor factorization results.

5.5.1 User Interface

Fig. 28 shows the user interface of *FacIt*, which consists of several views to support analytical tasks: (1) Model Inspection View (Fig. 28a) shows the comprehensive quality metric for model selection (T.1); (2) Pattern Projection View (Fig. 28b) gives an overview of patterns for experts to instantly understand the entire pattern space (T.2); (3) Along with Pattern Projection View, Pattern Detail View presents patterns at multiple scales from multiple perspectives (T.4); (4) Pattern Fine-tune Mode supports a set of interactions that allow experts to manipulate the patterns as well as item relationships, so that they can find the ones that align best with their domain knowledge (T.6); (5) Pattern Query Mode provides a query panel and item bars that experts can use to query patterns based on their specific interests, and then encodes the level of the relevance in Pattern Projection View; (6) Pattern Comparison Mode updates the pattern overview and pattern detail view to facilitate the comparison of the patterns.

Usage Scenario. We revisit the example of NBA shot data mentioned in Section 3 to illustrate how the different views of *FacIt* work together. Users start by selecting a model with a specific rank in Model Inspection View, and identify plausible ranks for further investigation. Once a rank is selected, Pattern Projection View provides an overview of patterns with different visual cues. Based on these, the users can immediately locate patterns that are dominant, isolated, or informative. When the patterns are not aligned with users' domain knowledge, the system enables users to fine-tune the patterns based, for example, on their knowledge of NBA players and their shooting patterns. In Item Projection View, users can move the relative positions of the items to incorporate their knowledge into the analysis. For example, one could move Stephen Curry closer to Klay Thompson if they have prior

knowledge that the two players have similar shooting tendencies. To specifically look into certain items, patterns, and their relationships, users could directly search for patterns that are relevant to particular items (e.g., shot patterns of LeBron James in overtime), by issuing a query through Pattern Query Mode. The results are displayed in Pattern Projection View. The query book supports tracking query history, allowing users to revisit searches, whenever they want to compare relevant patterns to different queries in the Pattern Comparison Mode.

5.5.2 Model Inspection View: Setting the Proper Rank

The model inspection view visualizes the summary statistics of tensor factorization models with different rank selections with a set of line charts (Fig. 28a). We present two quality measurements with respect to the degree of fit (*normalized reconstruction error* and *model fit*), one metric corresponding to the model sensitivity to the initialization (*model stability* [233]), and two metrics related to the interpretability of the model (*normalized entropy* and *sparsity*). To increase the robustness of the measurement for each rank, we perform five independent runs of the Tensor Factorization and report the mean and standard deviation in the line charts.

The system allows the users to efficiently consider each metric when selecting the rank. We present a rank suggestion in the leftmost line chart that weighs in users' priority order of different metrics, for the sake of transparent rank setting. When users mouse over a point in one line chart, the corresponding points for the same rank would be highlighted in all other charts for users to view. To support the confirmatory examination of a particular rank, once users click on the point associated with a rank, the system presents the corresponding model for review.

5.5.3 Pattern Projection View: High-Level Exploration

The pattern projection view provides an overview of the patterns. Each pattern is presented in a novel form of flower glyph that encapsulates its key information, descriptors, and relations to other patterns.

Projection View. This view provides an overview of the relationships between patterns in two dimensional space (Fig. 28(b)). Since each pattern is jointly described by multiple descriptors, we use a multi-view extension of Multi-Dimensional Scaling (MDS) [218] to map the patterns to a 2-D space. As a result, the pattern projection view illustrates the pairwise relationship of the patterns (i.e., similar patterns are located close to each other).

Pattern Glyph. We present the design of pattern glyphs that effectively summarize the following information (Fig. 28(b1)):

- Pattern Dominance. Analogous to PCA, where each component is associated with an amount of variance explained, we can also rescale the columns of each factor matrix to be unit length, and absorb the scalings into λ_r for each pattern r.
- Descriptor Informativeness. Given a descriptor $\mathbf{u_r}^{(m)}$ with m_i set of items, we first compute entropy $entr_r^m$ of $\mathbf{u}_r^{(m)}$ and use it as a proxy of its informativeness.
- Descriptor Similarity. We use $\bar{\mathbf{u}}^{(m)}$ to denote the distribution of the m-th descriptor averaged over R components. The similarity between the m-th descriptor of the r-th component to $\bar{\mathbf{u}}^{(m)}$ is calculated based on a spearman rank correlation due to the non-normal distribution.

For an efficient exploration of the patterns, we wanted to visualize high-level information about each pattern with a specially designed glyph. The multi-dimensional nature of the pattern suggests an intrinsic design of a flower with *petals*, one for each descriptor. We encode each petal using an *ellipse* and rotate the ellipses so that all petals take up the entire circle (360°) as shown in Fig. 27(b1). In addition, we apply a diverging color schema to differentiate different descriptors. We encode pattern dominance λ_r with the saturation of the outer circled area (outside of the ellipses). In this way, the more saturated the color, the larger variance explained by the corresponding pattern. While fixing the width of the m-th ellipse, we use its height to encode the value of entropy $entr_m$ in the m-th descriptor, where: a petal with a slim ellipse indicates that the descriptor has a large value of entropy, suggesting that the corresponding distribution is somewhat balanced over the entire set of items; otherwise, the corresponding distribution is dominated by a small set of items. We encode the similarity of the descriptor to the average distribution using the color saturation of

the petal. When users mouse over the pattern in the pattern overview, a tooltip is displayed that describes high-level summary statistics about the pattern. Users can click to select a pattern in the pattern overview, and the system shows its details (Section 6.3).

Design Alternatives. Over the course of working with our experts, we proposed alternative designs, such as Fig. 28(b1b) and Fig. 28(b1c). In Fig. 28(b1b), the curvature of the petal represents the informativeness, meaning that a curved petal indicates a more focused distribution while a round petal indicates more balanced distributions. The similarity in this design is double-coded by the size and color saturation of petal. We decided to use an alternative since experts were concerned about the effectiveness of curvature in differentiating different levels of entropy values. In Fig. 28(b1c), the circles represent the descriptors while the radius indicates the level of informativeness and the color saturation indicates the similarity. While this design appealed to experts more than the curvature-based design, circles of descriptors with small informativeness become extremely small. As a result, the similarity of the descriptor (the color saturation) became too difficult to read.

In the course of prototype designs, we discussed the need to automatically prune patterns based on the entropy of the descriptors. This requires users to provide two additional sets of parameters. The first is the threshold of informativeness, e.g., $\hat{entr}^m$, for each descriptor m, as the distribution of informativeness for each descriptor can vary. The second one is that the number of descriptors having informativeness below the corresponding threshold $\hat{entr}^m$, so that we know the pattern bears little informativeness. However, we dropped this design because it often complicated experts' interactions with the system. In fact, while our experts were interacting with the system, they found that, with the final petal design, they were able to quickly spot patterns with various informative characteristics, given the truthful presentation of all the patterns.

5.5.4 Pattern Detail View: Interpreting the pattern

Pattern Detail View contains two coordinated views, one for the quantitative distribution and the other one for the qualitative narrative of the pattern.

First, we provide a circular bar chart design to present the quantitative details of each

pattern (Fig.28(c1)). We use circular design for three reasons: 1) enabling space-efficient visualization by locating the pattern projection view inside its circular layout; 2) maintaining a consistent design between the visual representations at high-level (patterns as glyph) and low-level (items as bar in the circular pattern descriptor details), where the color and orientation of each dimension's visual encoding corresponds to each other; 3) expediting the exploration process, such that users can easily transition between the Pattern Overview and Pattern Detail View. This view consists of multiple circular bar charts, each of which indicates each descriptor. Each Item in a descriptor is represented as a bar corresponding to its value.

Second, we provide a Treemap View for users to qualitatively examine pattern narratives (Fig. 28(c2)). The Treemap provides compact and space-filling displays of hierarchical information, which summarizes the nested nature of the pattern, descriptors of the pattern, and items of each descriptor. Each small rectangle within the Treemap represents an item, with its value represented as its relative size, and its membership of descriptor represented as its color.

5.5.5 Pattern Fine-tune Mode

Following the same flavor of human-in-the-loop design as [35, 51, 52, 103, 125], *FacIt* was designed to allow users to progressively guide the factorization results towards their understanding of the data. According to the design requirements, the system should allow users to provide feedback to the patterns to directly refine the model and also to refine the items as an indirect approach to update the model.

Pattern Refinement. This interaction enables the users to delete and merge patterns so that they can directly give feedback to the model. The interaction for pattern deletion is straightforward. After selecting one pattern from the pattern overview, users can click on the "delete" button to remove the pattern from the space. Users can select two patterns and click on the "merge" button to combine them into a single one. Our approach to generate this new pattern is as follows. Given two patterns i and j, the values for each descriptor d of the newly merged pattern is the sum of each corresponding item value in d_i and d_j. For instance,

suppose two patterns in the coupon purchase dataset have gender descriptors of (0.4, 0.6) and (0.3, 0.7), respectively. When merging these two patterns, the gender descriptor of the resultant pattern would be set to (0.4+0.3, 0.6+0.7), followed by a normalization so that the sum becomes 1. We perform the same operation for each of the descriptors to obtain the reference matrix $\mathbf{U}^{(m')}$.

Item Refinement. Item refinement allows users to provide feedback to the factorization results in an indirect manner. However, our experts found it more straightforward and efficient, because it allows them to best utilize their domain knowledge to validate and improve the quality of factor matrices. To support this interaction, *FacIt* provides the *Item Projection View* for each descriptor. In this view, each item is represented as a circle with the color indicating its descriptor and label indicating its name. The position of the item is calculated based on MDS projection. Users can examine the relationships between the items. In addition, the system supports drag and drop for the item circles so that users can move and update the position of the items. Once users finish refining and click on the "update" button, the system takes the updated positions of the items and constructs a pairwise distance matrix $\mathbf{P}^{(m')}$ that reflects users' understanding of the data.

5.5.6 Pattern Query Mode

Pattern Query Mode allows users to efficiently locate patterns that match with their explicit points of interests. Fig. 28(e1) and (e3) presents two alternatives ways for users to issue queries:

Query Panel. We develop the query input box for each descriptor to allow users to input items of their interest. A query may have a single item from one descriptor or several items from multiple descriptors. Once a query is made, the most relevant patterns are retrieved and ranked based on their *relevance* to the user query. Given a query $Q = \{q_1, q_2, \ldots\}$ that consists of multiple queried items, the relevance of a pattern r to the query Q is given as: $rel(r) = \prod_{m \in M_Q} \prod_{i \in I_Q^{(m)}} \mathbf{u}_{ir}^{(m)}$, where M_Q are all modes (dimensions) involved in the query, $I_Q^{(m)}$ are the set of items involved in the query from the m-th dimension, and $\mathbf{u}_{ir}^{(m)}$ is the descriptor value for the i-th item from the m-th dimension with respect to the r-th

component. Once an item is added to the query, query is displayed as removable tags. Users can click the "close" button to remove the items from the query or the save icon in the query panel to save the query.

Item Bars. The item bar is placed in the inner ring of the circular bar in alignment with item circular bar chart. Users can trigger a query by simply selecting items of interest in the item bar.

To help users keep track of queries that they have performed, we add the support of a query book for users to bookmark queries that they would like to quickly retrieve later. Upon the query being issued, the system presents the most relevant patterns to the query as follows: 1) the set of relevant patterns is highlighted in the pattern overview with the rank of relevance added to the center of the pattern glyph; 2) at the same time, the pattern list view shows the ranked list of patterns based on the relevance score.

5.5.7 Pattern Comparison Mode

The system provides both high-level comparisons and details-on-demand comparisons. To support comparative analysis between the patterns, we use the Pattern Projection View and Detail View. Once a pattern is selected, the system updates the color saturation of the petals for the rest of the patterns, according to their similarity to the selected pattern. When users select any two patterns, the two bar charts are aligned such that 1) each item bar from the two charts is adjacent to each other, and 2) items are re-ordered based on the difference between the two patterns for the same descriptor by manipulating *superposition* and *explicit encoding* [68]. In this way, each descriptor in the circular bar chart pushes the discriminative items to the outside and keeps common items near the middle to ease analysis of common and discriminative items.

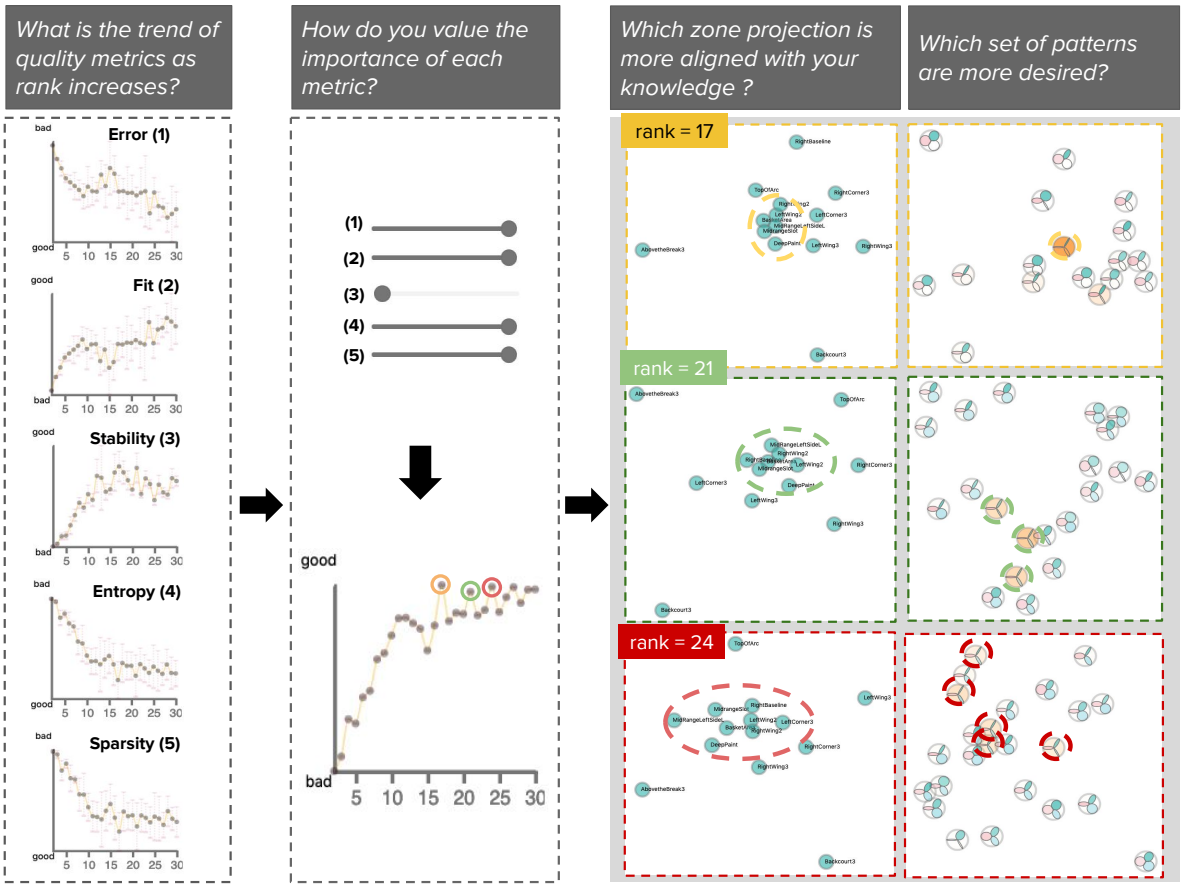

Figure 29: **NBA Shots Analysis: Model Inspection View**. Once `Kevin` observed the trend of the quality metric with respect to rank, he specifies the weight of contributions to the rank suggestion for plausible ranks. `Kevin` further clicked on the points to quickly validate the corresponding model outputs.

5.6 CASE STUDIES

Our case studies were developed by our domain experts and are to provide real-world examples that demonstrate how the system supports users in different applications of multi-aspect pattern discovery.

5.6.1 Model Inspection with NBA shots data

In this section, we work with our domain expert in sports analytics, `Kevin`, and uses an NBA shot dataset from the 2014-15 season to illustrate the system's model inspection capability.

Determine Rank Among Candidates. `Kevin`began by examining the trend of convergence with different objective rank metrics in the Model Inspection View (T.1.1). At this time, his analysis aimed at better *interpreting* the patterns rather than obtaining the *optimal* result; this led him to set the model stability to 0 and others to 1 (T.1.2). After adjusting the weights, the rank suggestion chart suggested three plausible rank candidates that have relatively high suggestion scores, which are 17, 21, and 24 (highlighted in orange, pink, and purple circles in Fig. 29). He clicked on those three dots in the rank suggestion chart sequentially to closely inspect the model with corresponding rank in the Circular View (T.1.3). He observed that as the rank increases, the model is distributed among multiple patterns, not dominated by just one (T.2.2). He was interested in verifying the model correctness by understanding the resulting item projection in the *zones*. He found that, with rank 21, the two-point zones are better clustered together when compared to the results of rank 17 and 24. After using the interactive model inspection process, `Kevin` was convinced that the shot data can be properly decomposed into 21 patterns.

Explore Crunch Shots. Once the model was selected, our analyst was interested in looking for "crunch time" shot patterns. Crunch time shots are ones that players take when the game is on the line, i.e., in over time while the score difference is small. He was particularly interested in comparing the representation of two players in this pattern, Stephen Curry and James Harden, both on top of the NBA's MVP list in 2014-2015. To do so, he clicked on the item bar that corresponded to quarter 5 and the item bar that corresponded to Stephen Curry (T.3,1). A set of relevant patterns was highlighted in the Pattern Projection View (T.3.2). After inputting a corresponding query for James Harden, our analyst selected two patterns for comparison, one from the most relevant patterns from each query. Fig. 30 shows the comparison of the two patterns (T.5.2). The most immediate difference is in shot locations, where the yellow pattern (Curry) tends to take more shots from the right-wing, followed by the deep paint, while the purple pattern (James Harden) tends to shot from the

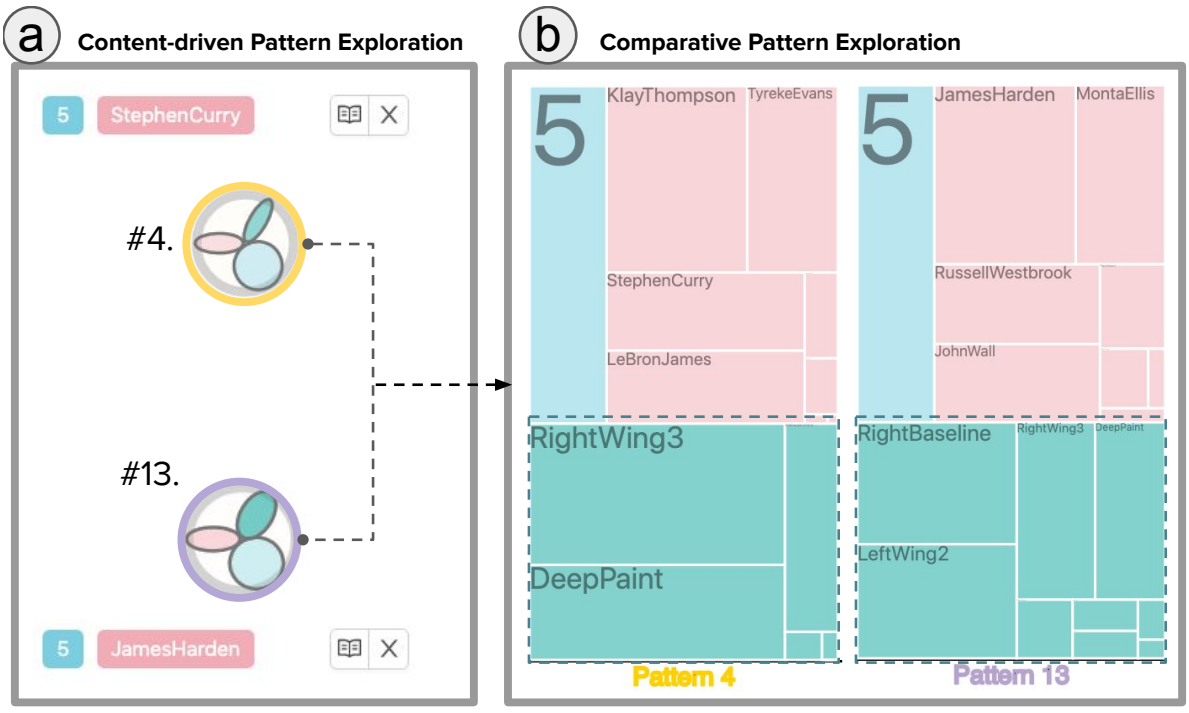

Figure 30: **Crunch Shots By Stephen Curry and James Harden**. Once `Kevin` identified the most relevant patterns to Stephen Curry and James Harden in overtime, he clicked on these two patterns to reveal their comparisons in detail.

left-wing and right baseline for two points. Compared to conventional match statistics, this pattern analysis from multi-aspect data provides more valuable insights as it further drills down behaviors associated with different aspects.

5.6.2 Model Fine-Tuning with Customer Behavior Data

In the second usage scenario, we work with our domain expert in business analytics, `Alice`, on a 5-dimensional coupon purchase dataset to demonstrate the capability of model fine-tuning based on users feedback. `Alice` wanted to find informative coupon purchase behaviors. She went through two iterations to optimize the result: (1) merging and deleting patterns, and (2) manipulating the relationships between items. She was able to identify interesting patterns with customized result.

Merge Similar Patterns. After fixing the rank at 25 based on the Model Inspection View, she started to refine the factorization results. Among the patterns that were clearly clustered in two sets (T.2.1), she noticed that two patterns located in the left of the projection view, shared almost identical ellipse shapes. She verified their similarity by selecting one of the patterns, which recolored other patterns by their similarity to the selected one she found that the two glyphs represent similar patterns in distributions of gender and age, and the category dimensions dominated by *Food* and *Delivery Service* (Fig.31ⓐ) (T.5.1). To keep patterns discriminative, she decided to merge (T.6.2) those two patterns (Fig.31ⓑ).

Delete Uninformative Patterns. As `Alice` continued her refinement, she found that one pattern has the thinnest ellipses in almost all descriptors and believed this pattern carried little informativeness (Fig.31ⓒ) (T.4.1). She clicked on the pattern and confirmed with the circular bar charts that the pattern had an almost uniform distribution in all descriptors except for the valid period (T.4.2). She decided that this pattern would not be informative in further exploration and deleted it (T.6.1). She then clicked on the update button to update the model.

Update Item Projection. In iteration #2, `Alice` focused on examining the item projection view. By examining this view, she was able to find relationships among items: For *Genre*, many genres formed a dense cluster while others were more isolated (Fig.31ⓓ). She noticed one item in the cluster, *Lesson*, has a purchasing intent that is clearly different from other categories. Then, she was able to incorporate her knowledge of purchasing intent, which was not explained by the dataset itself, by moving *Lesson* away from other items. At the same time, She moved *Spa* and *Relaxation* closer to the yellow cluster to teach the model that these items are more similar to items in the cluster (T.6.3). She then clicked on the update button, and was satisfied that the fine-tuned factorization results were more aligned with her knowledge (Fig.31ⓕ).

Informative Patterns in Coupon Purchase. When our analyst was presented with the final overview of the patterns, she was interested in locate the informative ones. She quickly located two patterns—pattern #12 and pattern #19 (Fig. 32), because both of them have the largest number of rounded petals (T.4.1). She clicked on these patterns to see their details in the Treemap View (T.4.3). While pattern #19 (in yellow) shows the purchase

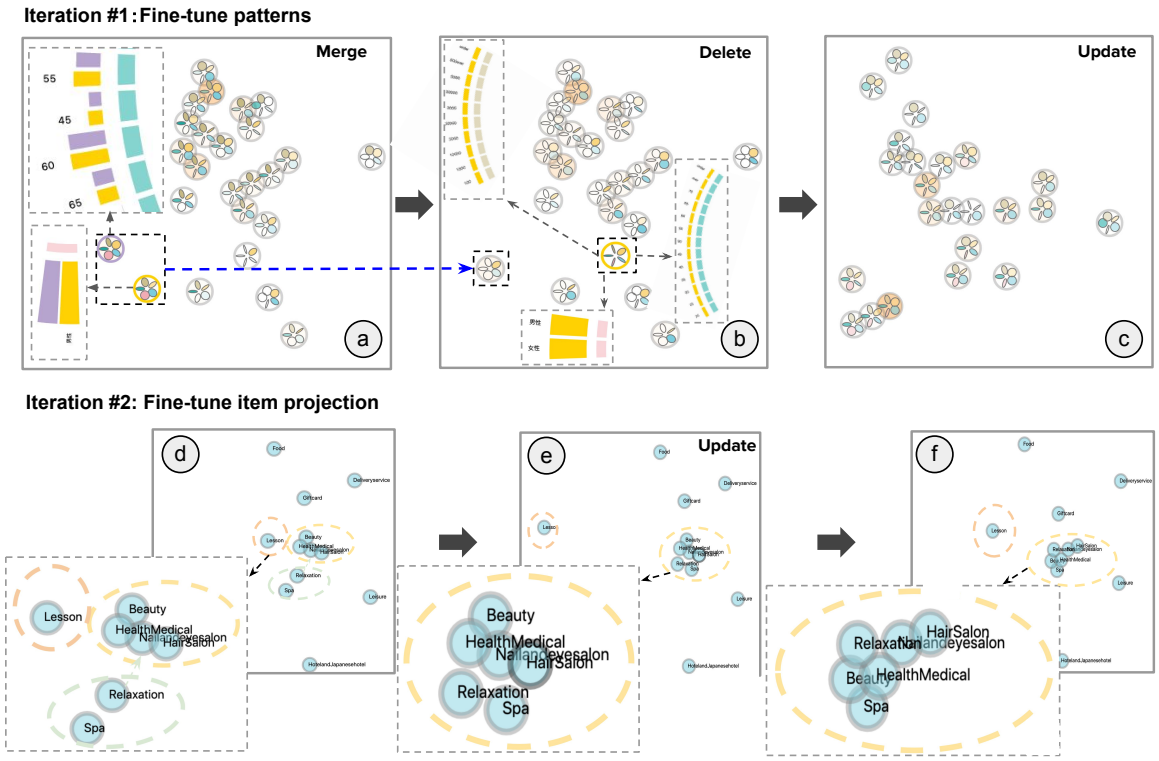

Figure 31: **Customer Behavior Analysis: Interactive Pattern Fine-tuning**. ⓐ`Alice` merged two identical patterns; ⓑ She further deleted a pattern that bears little informativeness; ⓒ The model updates to incorporate her feedback; ⓓ The system presents the relationships between the coupon genres; ⓔShe updated the positions for certain genres; ⓕThe system updated the model to reflect her feedback.

behaviors of food related coupons from male and female seniors over 60, pattern #12 (in purple) describes interesting behaviors of young, female customers for the relaxation and spa related coupons.

5.6.3 Pattern Exploration with Policy Adoption Data

We present a case study of policy adoption analysis to illustrate the effectiveness of *FacIt*. In this case study, we discuss the pattern analysis of policy adoption data in US state

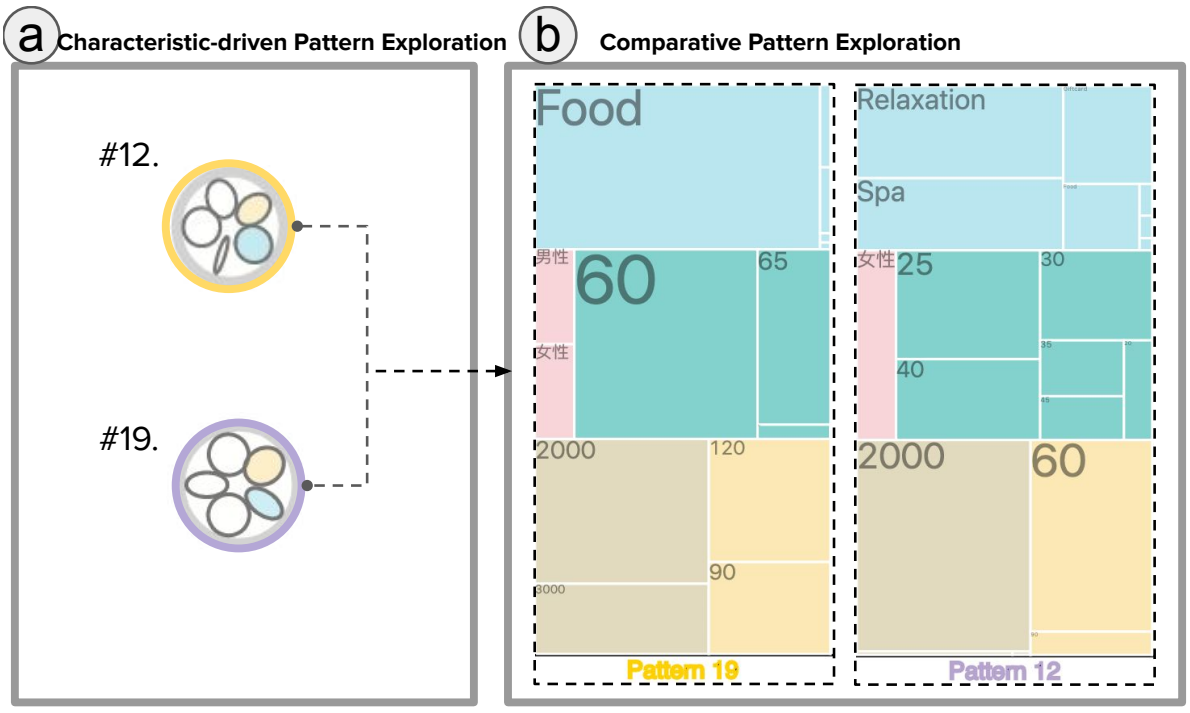

Figure 32: **Informative Patterns from Coupon Purchase Data**. `Alice` identified the two most informative patterns based on their pattern glyphs (ⓐ) and clicked on them to see their details (ⓑ).

politics since the 1990s. `Yvonne` is a policymaker in the department of public health in the Pennsylvania state government. She is now working to propose health insurance legislation. She has three questions to address: (1) What are the dominant adoption patterns of health-related policies? (2) Can we further decompose the dimension of those patterns into specific health-related topics? How are the dominant patterns different from each other? (3) Are the political interests of states different? How are they related to a state's characteristics (e.g., liberal or conservative)?

Content-driven Pattern Exploration. She started by querying health policy. After selecting "Health" in the subject query box (T.3.1), she found that the system highlighted a set of patterns that are related to "Health" policy using the color saturation in the pattern glyph in the Pattern Projection View (T.3.2). She focused on exploring the four most relevant

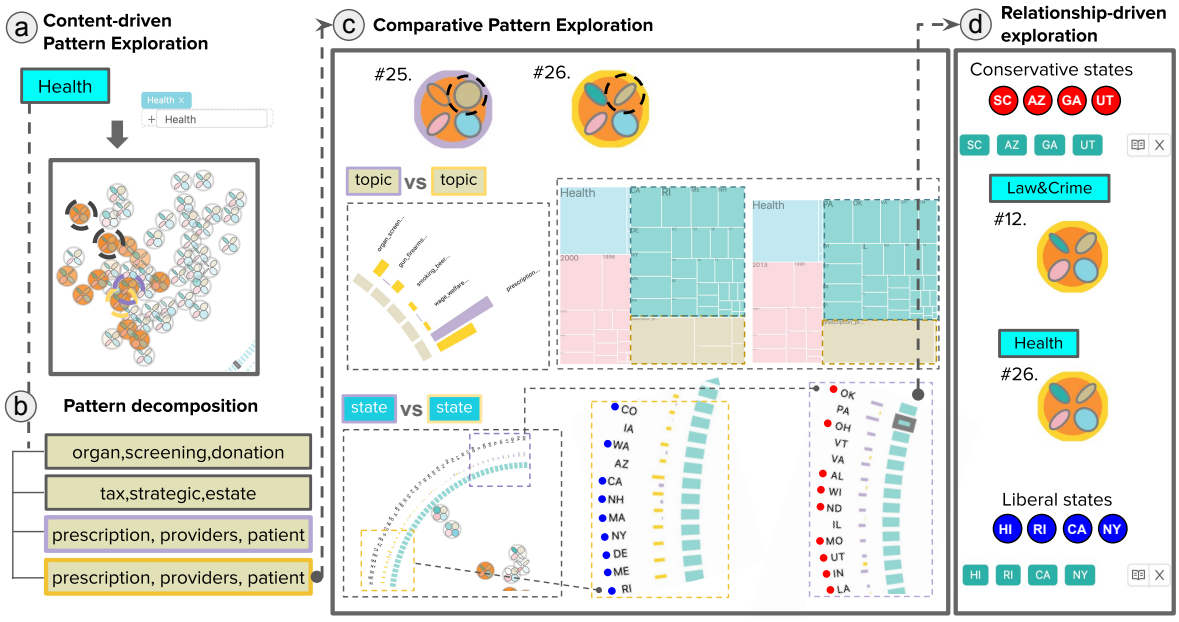

Figure 33: **Policy Adoption: Pattern Scrutinization**. ⓐ Yvonne first queried the "Health" category to detect the most relevant patterns. ⓑ She found that the topic dimension decomposes patterns health-related pattern into different political agendas, two of which fit into her interest. ⓒ When she selected and compared those two patterns, they had different distributions of topic and state. She especially noticed that states with the opposite ideology are dominant actors in the patterns. ⓓ She further explored the relationship between the most conservative and liberal states by querying each of them. The system showed her the difference in terms of which category is most relevant.

patterns with the greatest relevance scores (Fig. 33ⓐ). When she looked at the dominant topics of these patterns, she noticed that they were characterized by different topics (Fig. 33ⓑ), which helped her analyze how health policies can be decomposed into policy agendas. Among them, Pattern 25 and Pattern 26 were especially of interest to her for two reasons: (1) The dominant topic of each is "prescription, patient, provider", which directly relates to her current interest (health insurance policy), and (2) The topical distributions of two patterns were different, Pattern 25 was dominant by a few topics and pattern 26 was spread across a higher number of topics (T.5.1).

Comparative Pattern Exploration. She began to explore two patterns in detail (Fig. 33ⓒ). In the dimension of states in the Treemap View, she noticed that Pattern 26 was mainly driven by liberal states, while Pattern 25 was mainly driven by conservative states (T.4.3 and T.5.2). She also found that Pennsylvania was the second most dominant in Pattern 25 among other dominant conservative states. In the topical dimension, Pattern 26 was all about "prescription, provider, patient" while Pattern 25 was distributed across multiple items with two dominant topics, "prescription, provider, patient" and "organ, screening, donation". While she was exploring the patterns, those ideological differences drove her to come up with another question: "How do policy agendas compare in liberal and conservative states?".

Relationship-driven Pattern Exploration. Now, she was interested in querying states by their ideology and analyzing relevant patterns. First, she identified the most liberal and conservative states. According to [19], the state ideology score in 2017 indicated that CA, RI, HI, NY were the most liberal states, and SC, AZ, GA, UT were the most conservative states. Since *FacIt* was able to issue multiple items in one query, she made two separate queries, one for each group of four states (T.3.1) and saved them to the query book (T.3.3). Interestingly, the two queries of liberal and conservative states resulted in different policy topics with different patterns (Fig. 33ⓓ). For liberal states, health-dominant Pattern 26 was the most relevant one. On the other hand, Pattern 12, mainly driven by "Law and crime", was the most relevant pattern topic in conservative states.

By combining these analyses, she learned patterns in health-related policies in detail, and could figure out in which states she could most likely make an impact.

5.7 DOMAIN EXPERT INTERVIEW

We invited the domain experts that collaborated with us to use and evaluate *FacIt*. These experts were selected based on two criteria: (i) they must be familiar with the domain of the data; (ii) and they must have experience of using tensor-based analysis to analyze multi-way association patterns. We performed an in-depth interview with each expert one-on-one. Each

interview lasted about 1.5 hours, and consisted of three sessions: (1) interface explanation and initial feedback (30 minutes): we introduced experts to the dataset and key modules of *FacIt*, and collected feedback on their first impressions; (2) using the system to explore patterns (30 minutes): a session where experts used *FacIt* to explore tensor factorization, and (3) a semi-structured interview (30 minutes): a post-study interview session where experts discussed the usability and suggested improvements. The rest of this section presents their aggregated feedback.

Visual Design: Experts were particularly impressed by the design of pattern presentation. They found that the pattern presentation in *FacIt*dramatically improved their efficiency when understanding, comparing and recognizing meaningful patterns. One of the experts told us, "It usually takes a long time for my colleagues and me to examine the patterns,..., before locating meaningful ones, especially when the number of them is very large". With the help of pattern circular view and projection view in *FacIt*, they felt it was "much easier to gain a comprehensive overview of patterns and understanding pattern relationships". They highly valued the design of the pattern glyph. They explained that, "the information jointly encoded in the glyph, such as pattern dominance, descriptor informativeness, etc. is really helpful to understand a pattern". Moreover, experts approved of the functions of selecting patterns for comparison and highlighting similar and discriminative items. All of these features make pattern comparison easier and more intuitive.

System Interaction: Experts felt that several interaction tools supported by *FacIt*were quite useful. With the help of those tools, they were able to easily incorporate their prior knowledge, feedback, and specific target into the final presentation of patterns. First, all experts appreciated having a model inspection module in *FacIt*. They pointed out that their pattern exploration process with tensor factorization usually starts by selecting a proper rank. Although they have data fitting evaluation metrics to consult with, they agreed that "the whole process takes a considerable amount of time and effort because this rank is not all about fitting to the data, but rather finding a set of interpretable patterns." With the *FacIt*'s inspection module, one expert reported that "the rank selection then becomes a trade-off problem among a transparent set of objectives", which makes the process much easier and faster. Second, the feedback-based model fine-tuning was very well received by our experts.

They confirmed that most of the time they all had some prior intuition or domain knowledge before they leveraged any tool to explore patterns. Prior intuition sometimes helped them locate meaningful patterns, but was unreliable, leading to unreasonable discoveries. One of the experts emphasized that "this interactive feature will be useful and phenomenal." He confirmed that "it visualizes patterns after being re-tuned with my prior intuition; this speeds up the process of locating meaning patterns when my intuition is correct, and helps me recognize much earlier if my intuition is far from the fact." Third, the experts gave especially positive feedback to the interactive pattern query module, where they "could explore patterns relevant to an explicit set of interests." Our experts tried different combinations of queries. "[I] needed to go through each component to find [ones] that are most relevant to my interests," said by one expert. He also commented, "This module significantly speeds up this process". He particularly appreciated the query book because it allowed him to quickly switch between visualizing of relevant patterns from different queries, allowing him to compare them more efficiently.

System Usability: According to the experts' feedback, they were satisfied with *FacIt*and considered it a comprehensive visualization and analysis system that fulfilled their requirements of understanding, exploring and interpreting patterns from a multi-aspect real-world dataset. For instance, the experts were confident that the system could be effectively applied to areas not limited to: (1) semantic analysis of knowledge base composed of tensors like (subject, verb, object), (2) community detection in multi-view, large-scale social networks, where each view corresponds to an aspect in tensor, and (3) discovery of road spatio-temporal relationships from traffic datasets, which play a critical role in determining traffic management strategies. The experts also expressed that the system could be efficiently and effectively applied not only to search for meaningful patterns, but also to generate new patterns with users' domain knowledge as input. The experts believed this would be extremely useful for researchers who want to verify their intuition with pattern visualization or hope to incorporate their domain knowledge into pattern generation.

Improvements: Although our experts agreed that *FacIt*was easy to use in general, they suggested some improvements and new features: (1) Domain-specific visual design: While our experts understand *FacIt*'s value as a generic visualization tool for multi-aspect

data in different domains, they suggested the system could use domain-specific visual design to make patterns more intuitive to users, e.g., encoding descriptors with makers whose shape and color have more semantic meanings in the corresponding domain. (2) Comparing different modules: All the experts highly valued the feature of interactive feedback-based model fine-tuning. However, they mentioned that it would be more convenient to compare patterns generated with different intuition. If the system could retain model results and allow experts to compare multiple models, experts could verify the validity of their intuition. (3) Active feedback collection: The experts suggested that it would benefit the pattern exploration process if feedback collection was two-way instead of one-way. Currently, only users of the system are allowed to re-tune model results and update pattern presentation. Two-way feedback collection would allow the system to actively collect feedback from users for parts of results it has low confidence in. The system could crowd-source and store the feedback from different users and update the default pattern display.

5.8 SUMMARY

In this chapter, we present *FacIt*, a visual analytic system for Tensor Factorization. The system is built to meet the common requirements of real-world applications, such as model selection, model refinement, and results scrutinization and interpretation. We have developed a suite of model scrutinization and inspection tools to empower the model selection process. A novel weakly semi-supervised tensor factorization algorithm is proposed to allow human-in-the-loop tensor factorization discovery. In addition, we provide an interactive design that caters to experts' different exploration strategies, such as characteristics- and content-driven pattern discovery. The effectiveness and usefulness of *FacIt* has been evaluated in usage scenarios across different domains, followed by in-depth interviews with domain experts.

6.0 DISCUSSION, CONCLUSION AND FUTURE WORK

Chapter 3, Chapter 4 and Chapter 5 incrementally build up to answer each of the three research questions posed in the Introduction (Chapter 1). In this chapter, I will review my framework towards interpretable tensor factorization for multi-aspect data. I will review the studies that demonstrate the effectiveness of the framework and further discuss their implications. I first provide the conclusions and contributions of this dissertation by answering each of the three research questions. Then, Section 6.2 presents a comprehensive discussion of the results. After that, I talk about the limitations of this dissertation in Section 6.3. Finally, Section 6.4 envisions potential future research topics.

6.1 CONCLUSION & CONTRIBUTIONS

6.1.1 Conclusions

Recent performance improvements in supervised learning call for the improvement of the interpretability of such models and their results. Therefore, much research has proposed ways to facilitate the understanding and explanation of models. However, considerably less attention has been given to the development of interpretability in unsupervised settings. This dissertation takes the initiative to look at the interpretability under the umbrella of unsupervised learning, and we propose a M^3 framework towards the interpretability in unsupervised pattern mining. We select multi-aspect data because an increasing amount of data generated has various multi-aspect characteristics. We use tensor factorization to demonstrate the proposed framework since it is one of the most popular techniques for uncovering

patterns in multi-aspect data.

The need to interpret for unsupervised mining is critical because of several challenges. 1) Mining with the mismatch between human information need and reconstruction errors. Simply applying an off-the-shelf mining toolkit does not respect the particular information need of the users. 2) Mining with insufficient evaluation criteria. Current evaluation schemas undergo either qualitative examination of outputs (e.g., topic modeling) or using downstream tasks as a proxy to measure mining quality (link prediction in graph representation learning). 3) Mining with the mismatch between experts' domain knowledge and data-driven models. Data can be noisy. Even if a model is tuned to the users' information need, the results may not readily translate to something domain experts' can agree upon. 4) Mining with the mismatch between underlying multi-aspect pattern and human understandability. Patterns from multi-aspect data require its interpretation simultaneously from multiple perspectives. Different presentations of a pattern can vary in experts' ability to understand them.

These challenges stimulate this dissertation. A M^3 framework of pattern discovery from multi-aspect data was proposed to address each challenge. To ease the mismatch between human information need and reconstruction-oriented factorization, we propose the multiplex pattern discovery component. In this component, the information need is operationalized through a regulative tensor factorization model that is tuned to users' information needs. To ease the evaluation of patterns from tensor factorization, we design a multifaceted pattern evaluation schema, where patterns are validated from multiple perspectives: *quality*, *validity*, and *utility*, where quality stands the overall quality of tensor factorization, the validity suggests how well the patterns can be explained by the experts' domain knowledge, and the utility evaluates the applicability of patterns in downstream tasks. To further close the gap between human interpretability and interaction with the factorization process, we propose a visual analytic system as a united approach to simultaneously address all the challenges.

This thesis introduced three studies to build up the components of the interpretation framework and demonstrate its effectiveness. In the first study, we situated the requirement of a carefully crafted model to cater to the information need in an urban space in the aftermath of the major events. Compared to a participatory assessment of the impacts of events [176], there is a crucial need for a data-driven model to reveal the impacts quickly.

With the increasing amount of multi-aspect data becoming available, we formulate the information need as a contrasting pattern discovery problem given multi-aspect mobility data from before and after major events in the city. We design a collective tensor factorization model, *PairFac*, to uncover the shared phenomena and discriminative phenomena. *PairFac* takes multi-aspect data as input and split into two groups, before and after certain events. We apply *PairFac*in two case studies and demonstrate its capability to reveal persistent and changing mobility patterns following events of interest. For example, in our first case study, using data from the terrorist attacks in Paris of 2015, we see that activities around professional life and food venues experienced the fewest changes.

In the second study, we target the information need in the domain of understanding contrasting user behavior patterns in massive online courses (MOOC), with a particular interest in identifying the relationships between underlying behavioral patterns and performance outcomes. We propose a tensor-based learning method, *iDisc*, that discovers the common and discriminative learning patterns at multiple levels. Based on this, it projects users to a latent space (i.e. *embedding* for the downstream prediction tasks) to identify the association between the multi-way interaction of the features and the students' performance. The empirical studies with the dataset from different MOOC platforms have shown that *iDisc* yields promising results on the effectiveness and efficiency.

In our third study, we propose a visual analytic system, *FacIt*, to simultaneously address all the mismatches we identified with pattern discovery from multi-aspect data. The system is built to meet common requirements, such as model selection, model refinement, results scrutinization, and interpretation for its real-world applications. We develop a suite of model scrutinization and inspection tools that empowers the model selection process. A novel weakly semi-supervised tensor factorization algorithm is proposed to allow human-in-the-loop tensor factorization discovery. In addition, we provide an interactive design that caters to experts' different exploration strategies, such as characteristics- and content-driven pattern discovery. The effectiveness and usefulness of *FacIt* have been evaluated through usage scenarios across different domains, followed by in-depth interviews with domain experts.

6.1.2 Contributions

In general, I believe that there are several key contributions of this dissertation:

- An understanding of the challenges of interpretability in unsupervised pattern discovery. To the extent of my knowledge, while interpretability in machine learning has become an increasingly known issue, this dissertation is the first to explore the problems of interpretability in unsupervised settings.
- A framework to address the challenges of interpretability in unsupervised pattern discovery. By understanding of challenges, this dissertation provides the first attempt to create a framework of interpretable pattern discovery from multi-aspect data. The M^3 framework consists of three components to address the identified challenges: 1) multiplex pattern discovery to close the gap between human information needs and standard pattern discovery tools; 2) multifaceted pattern evaluation to validate patterns via multiple approaches; and 3) multipurpose pattern presentation to close the gap between patterns and human understandability, and allow experts to provide feedback in the loop of pattern discovery.
- A demonstration of the effectiveness of the proposed framework with three studies. The dissertation contributes three studies that are incrementally organized to demonstrate the use of the framework in solving real-world problems of interpretability where human information needs intersect with pattern discovery from multi-aspect data.

6.2 DISCUSSION OF RESULTS

My dissertation proposes a framework for interpretable tensor factorization for multi-aspect data. To formulate the framework, it presents three studies that incrementally address interpretability in the process of pattern discovery. Beyond the specific results of each study previously discussed, reviewing them as a whole could lead to key insights into a better design towards interpretable pattern discovery from multi-aspect data.

6.2.1 Multiplex Pattern Discovery to Ease the Mismatch Between Human Information Need and Naive Error-Based Optimization

To properly repair the mismatch, this dissertation argues that one plausible way is to understand the information need and design customized models beyond the standard pattern discovery process. The different information needs presented in this dissertation are all structured under the same idea of understanding the exact information need and conceiving the corresponding problem formulation with both reconstruction and human information need.

Our first study targets event impact analytics in the aftermath of disasters in a city. Compared to a typical participatory impact assessment, there is the need for an expeditious data-driven evaluation mechanism to present an understanding of the impact on the stakeholders in the city. Accordingly, we formulate a problem of contrasting pattern discovery in the mobility data, leveraging its multi-aspect nature. *PairFac* is designed to identify underlying mobility patterns, for understanding persistent and the changing patterns among them. Our second study tackles the information need to understand the behavioral patterns of users on MOOC platforms from different performance groups. In addition to recognizing the patterns that lead to different performance outcomes, there is also a need to explore patterns at multiple scales. To cater to the information need, we formulate a problem of multi-level discriminative pattern discovery from a pair of tensors. *iDisc* is an iterative framework that reveals the contrasting patterns at multiple levels. In our last study, we address the problem of generic tensor factorization. The multiplex pattern discovery works in such a way that it presents a comprehensive list of metrics. Users can tune the model directly based on their information needs, including sparsity, stability, and quality of reconstruction, in the model inspection tool of *FacIt*.

This connects to existing work in "model-based" interpretable supervised machine learning, where users may favor a revised model for the sake of being able to interpret it. For example, smaller models [17,78,85] or sparse models [254] are preferred over large, black-box models.

6.2.2 Multifaceted Pattern Evaluation to Mine Under Insufficient Evaluation Criteria

While existing unsupervised learning focuses on revealing underlying data patterns, there has not been a systematic way to evaluate these patterns. When evaluating of tensor factorization, one line of work focuses on validating patterns from domain experts' points of view (refer to survey paper at [7]). While pattern examination often leads to hidden insights in multi-way interactions, how it deepens our understanding of the data is unclear. Another line of work directly evaluates via applications of the patterns in downstream tasks (e.g., recommendations [21,97,175,182,185]). However, despite the success of such work, they still leave the users with a black-box model without explaining the underlying mechanism for the generation of recommendations.

Given the increasingly popular use of tensor techniques, we call for a multifaceted pattern evaluation, which considers quality, validity, and utility of the results: quality stands the set of metrics that evaluate the overall factorization performance, such as reconstruction error; validity indicates how well are the results aligned with experts' expectations based on their domain knowledge; and utility suggests the applicability of the results in downstream tasks, such as clustering, classification, or recommendations. In our first study, *PairFac* was able to generate a set of patterns that describe the impacts of major events in the city. However, it is clear unclear how the patterns can be used beyond explaining and examining what has happened. In our second study, we conduct an intrinsic evaluation to make sure the patterns from *iDisc* make sense to experts. In addition, we involve domain experts to qualitatively examine the utility of the patterns in a classification task. In our third study, *FacIt* first presents a comprehensive set of quality indexes. Then, the validity of the patterns is checked by the experts via directly examining them, and the utility of patterns is verified by inspecting the pairwise relationships between items based on the patterns.

We need to acknowledge that such evaluation schemas are not new to the field of tensor factorization. For example, Ho et. al [87] addressed the interpretability and predictivity of phenotypes discovered from multi-aspect data built from electronic health records. This echoes with our call for multifaceted pattern evaluation in both validity and utility.

6.2.3 Multipurpose Pattern Presentation to Overcome the Mismatch Involved Domain Knowledge and Human Understandability

To the best of our knowledge, this dissertation presents the first attempt to involve experts in the process of pattern discovery in a generic, multi-aspect setting, with novel pattern presentations and interaction mechanisms.

Tensor factorization has many applications in a wide range of domains, e.g., telecommunications [46, 196, 197], neuroscience [136, 143, 147], and data mining [206, 207]. However, few applications involve the domain experts in the process of pattern discovery. Our first two studies fall into the category of not utilizing experts' domain knowledge. It has become increasingly alarming to us how much of a gap there is between discovered results and results that experts can understand. To address this problem, our last study argues that having experts in-the-loop along with the thoughtful design of pattern presentation can lead them to explore better, interpret, and refine patterns.

Multipurpose pattern presentation features several novelties in the visualization design of patterns. First, it sits in an interactive visual analytics system, which allows experts to manipulate patterns and provide feedback to refine them. Second, the pattern presentation features high-level summary displays that empower efficient exploration and identification of patterns. Last but not least, the pattern presentation is enriched by both quantitative and qualitative visualizations that allow for detailed pattern examination on demand. We believe the multipurpose nature of pattern presentation elevates tensor pattern discovery to a transparent and effective process for human understandability, an interactive and responsive mechanism to build upon domain knowledge.

6.3 LIMITATIONS

Despite the above contributions and insights, I acknowledge that my dissertation has several limitations, which are discussed in the following subsections.

6.3.1 Limited Guidance in Pattern Evaluation

First, it is important to acknowledge the evaluation strategy in the three studies, especially for the first two studies are not results of rigorously following our proposed guideline for multifaceted pattern evaluation. In fact, it is through the reflections of these sequential studies and observations of the current practice in the field that we come to the realization of the deficit efforts in a thorough evaluation standard.

Multifaceted pattern evaluation suggests considerations when experts devise their evaluation strategy of multi-aspect mining, but with limited details on the exact guidelines, they follow to design experiments in a way that meets these considerations. Since the specific context, data, and task can vary, the reflection of each consideration can also be different. The quality aspect of the results consists of several quantitative measurements. The validity of the patterns is mostly done with manual inspections by the experts in real-world applications with exceptions of quantitative evaluation when the synthetic dataset is used in the experiments. However, the evaluation through validity is more complicated because the tasks involved vary. For example, *iDisc* evaluates the utility in a quantitative and prediction task while *FacIt* inspects the utility through a qualitative examination over pairwise item relationships in the embedding space of the items (e.g., player embeddings). In this regard, this thesis provided a limited understanding of specific evaluation suggestions, especially when evaluating the aspect of utility.

Besides, the guidance suggested might not be directly transferable to other unsupervised tasks. For example, multifaceted pattern evaluation for tensor factorization urges the consideration of quality, validity, and utility of hidden patterns. While it is a task to qualitatively examine the validity of the latent patterns from the tensor factorization, it is often not applicable to vet the continuous latent space as a result of other unsupervised tasks, e.g., auto-encoder, or PCA. This requires an appropriate realization of quality, validity, or utility. For example, Hsu et. al [89] propose a factorized hierarchical variational autoencoder that learns disentangled and interpretable representations from sequential data. The validity of the latent representations is examined by manually scanning the sequence level and segment-level variables, and their connections to sequence-level and segment-level attributes

of the speech.

6.3.2 Limited Context of Information Need

Multiplex pattern discovery from multi-aspect data takes the information need and formulates a customized regularization module in the objective function of the factorization model. I acknowledge that this dissertation illustrates the process and the result of analyzing for particular kinds of information need.

FacIt, looks at a generic tensor factorization model with the extended capability to incorporate experts' information needs, when such needs are not available upon the factorization. While most of the human information needs in multi-aspect data aim to reveal patterns from a single tensor (or coupled with heterogeneous data sources), we are interested in addressing the needs for pattern discovery from a pair of tensors. For example, *PairFac* takes the information need to understand contrasting patterns given a pair of multi-aspect data. *iDisc* takes it one step further with the assumption that contrasting patterns can reside at multiple levels, and a hierarchy-driven pattern exploration could aid the exploration process. At the heart of *PairFac* and *iDisc* is, the problem of contrastive pattern discovery from a pair of multi-aspect data. We focus on this problem because we believe it is the root analysis of many application domains, such as multi-aspect biomarker discovery in biology, anomaly detection in multi-aspect time-series. In the future, we would also like to investigate human information needs that are related to pattern discovery from a set of tensors. While these studies shed light on different principles of modifying standard factorization models, namely, collective, iterative, and interactive, divergent requirements of human interests should be discussed so that a suitable set of corresponding models can be designed.

6.3.3 Limited Usage Scenarios of Tensor Factorization

This dissertation tackles problems of tensor factorization in specific usage scenarios without considering the inclusion of auxiliary data upon the factorization. We target translating concrete information need in pattern discovery to operationalizable constraints in the objective function in the first two studies. Our last study presents a model that can incorporate

domain knowledge, should it become available. However, it still aims for a generic tensor factorization setting for experts to start with. Compared to advances in the area of data fusion within the framework of tensor factorization, the M^3 framework might come up short for experts who have the clear and well-structured domain knowledge to fuse into the factorization process. We believe that our M^3 framework can still be used to guide interpretable pattern discovery from multi-aspect data in such cases. Experts can benefit from multi-faceted pattern evaluation and multipurpose pattern presentation to better understand and communicate their results.

6.3.4 Limited Tasks in Unsupervised Learning

Unsupervised learning consists of a broad set of tasks for datasets, where labels are not explicitly available. Common problems that fall into this category include clustering, autoencoder, generative adversarial learning, and other latent variable models. This dissertation focuses on pattern discovery from multi-aspect data due to its increasing number of application areas. Although we have presented several studies as proof of concept for the proposed framework, it requires non-trivial efforts to generalize it to other unsupervised learning tasks. Different mining tasks might inherently present varying challenges that require an in-depth understanding of corresponding problem domains. The M^3 framework is proposed to address challenges in mining from multi-aspect data. We conjecture our framework could work with unsupervised tasks for the discovery of latent representations of some form that experts can comprehend, such as interpretable latent representations from autoencoder [89, 220]. However, the solutions might not be pertinent to other mining tasks where the latent states are rather difficult to interpret, such as generative adversarial networks or reinforcement learning.

6.4 FUTURE WORK

In the future, I would like to explore the following research directions.

6.4.1 Data Fusion in Tensor Factorization

As explained in Section 6.3.3, none of the studies within the scope of this dissertation have systematically assumed domain knowledge as priors of pattern discovery. The M^3 framework has been designed for novice users of tensor factorization and crafted in a way that minimum domain knowledge is required to kickstart pattern exploration.

As users become more comfortable with tensor factorization, their understanding of the problem and the corresponding solution space could be enriched by additional information they gather. Therefore, I am interested in exploring the framework for advanced use cases. Our experts could have domain knowledge in various forms of well-structured data that they can use to guide pattern discovery. For example, Acar et al. [4] propose a coupled matrix and tensor factorization framework and use data from multiple sources to aid the discovery of underlying data structures from the multi-aspect data. This triggers the understanding of the challenges involved in factorizing multi-aspect data, where multiple data sources jointly describe the data. For instance, given a tensor of customers' rating history over a set of items, specific interpretability challenges will arrive in the process of pattern discovery, with the availability of customer-customer friendship or item-category information.

The M^3 framework can potentially be tuned for pattern discovery in these scenarios, considering they use multiple data sources. In our work, multiplex pattern discovery devises regulative terms as proxies of information need. This multiplex pattern discovery could be extended to bridge the domain knowledge into the objective function in a similar way, as constraints of the model. Our current practice of multipurpose pattern presentation focuses more on the interpretation and exploration of patterns, rather than the multi-aspect data input itself. Given the multiple data sources concerning the targeted multi-aspect data, additional efforts would need to be made to understand the various data input and relationships among them. A relational hypergraph describing the relationships between data sources [121] could be a promising interactive channel for users to explore and exploit the domain knowledge in tensor factorization. On the other hand, multifaceted pattern evaluation requires additional considerations regarding the empowerment by domain knowledge, beyond the validity and utility of the patterns.

6.4.2 Generalization to Other Unsupervised Tasks

This dissertation focuses on one of the most popular tensor factorization techniques, CANDECOMP/PARAFAC decomposition. Among the alternatives, Tucker decomposition is also one of the most studied decomposition techniques, although it lacks the aid of interpretability in the results. Although the parameters involved differ from CP decomposition, we believe the M^3 framework is a promising start to understand the specific requirements of interpretable tucker decomposition from multi-aspect data.

With the advancements of computing power and the availability of massive amounts of data, recent work has explored beyond linear transformations of data to latent patterns and studied the potential of latent embedding as a result of non-linear processes. For example, neural collaborative filtering [82] replaces the inner product of the user and item embedding with a neural architecture that can learn an arbitrary function from data. Neural tensor factorization [235] uses a multi-layer perceptron structure for learning the non-linearities between different latent factors. While these works present the potential to leverage non-linear embeddings in their respective tasks, they also pose unprecedented challenges. Since such work is in the intersection of tensor factorization and deep neural networks, it would be interesting to explore frameworks that can simultaneously interpret latent factors from the multi-aspect data and the latent factors from a non-linear transformation process.

I am fascinated by the fact that we can use a more compressed and more abstract form of data to represent noisy data, possibly being massive in size. Tensor factorization is truly one of such processes. Beyond that, there are other unsupervised tasks that require the aid of interpretability. For example, neural networks, such as Autoencoders [123, 124] and Generative Adversarial Networks [70], learn efficient data codings from the data. It would be interesting to devise a framework towards interpretable latent representation learning from these neural networks.

BIBLIOGRAPHY

[1] Evrim Acar, Rasmus Bro, and Age K Smilde. Data fusion in metabolomics using coupled matrix and tensor factorizations. *Proceedings of the IEEE*, 103(9):1602–1620, 2015.

[2] Evrim Acar, Daniel M Dunlavy, and Tamara G Kolda. A scalable optimization approach for fitting canonical tensor decompositions. *Journal of Chemometrics*, 25(2):67–86, 2011.

[3] Evrim Acar, Daniel M Dunlavy, Tamara G Kolda, and Morten Mørup. Scalable tensor factorizations with missing data. In *Proceedings of the 2010 SIAM international conference on data mining*, pages 701–712. SIAM, 2010.

[4] Evrim Acar, Tamara G Kolda, and Daniel M Dunlavy. All-at-once optimization for coupled matrix and tensor factorizations. *arXiv preprint arXiv:1105.3422*, 2011.

[5] Evrim Acar, Evangelos E Papalexakis, Gözde Gürdeniz, Morten A Rasmussen, Anders J Lawaetz, Mathias Nilsson, and Rasmus Bro. Structure-revealing data fusion. *BMC bioinformatics*, 15(1):239, 2014.

[6] Evrim Acar, Morten Arendt Rasmussen, Francesco Savorani, Tormod Næs, and Rasmus Bro. Understanding data fusion within the framework of coupled matrix and tensor factorizations. *Chemometrics and Intelligent Laboratory Systems*, 129:53–63, 2013.

[7] Evrim Acar and Bülent Yener. Unsupervised multiway data analysis: A literature survey. *IEEE transactions on knowledge and data engineering*, 21(1):6–20, 2008.

[8] Harshavardhan Achrekar, Avinash Gandhe, Ross Lazarus, Ssu-Hsin Yu, and Benyuan Liu. Predicting flu trends using twitter data. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 702–707. IEEE, 2011.

[9] Ardavan Afshar, Joyce C Ho, Bistra Dilkina, Ioakeim Perros, Elias B Khalil, Li Xiong, and Vaidy Sunderam. Cp-ortho: An orthogonal tensor factorization framework for spatio-temporal data. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, page 67. ACM, 2017.

[10] Rakesh Agrawal, Behzad Golshan, and Evangelos Papalexakis. A study of distinctiveness in web results of two search engines. In *Proceedings of the 24th International Conference on World Wide Web*, pages 267–273. ACM, 2015.

[11] Genevera Allen. Sparse higher-order principal components analysis. In *Artificial Intelligence and Statistics*, pages 27–36, 2012.

[12] Miguel Araujo, Spiros Papadimitriou, Stephan Günnemann, Christos Faloutsos, Prithwish Basu, Ananthram Swami, Evangelos E Papalexakis, and Danai Koutra. Com2: fast automatic discovery of temporal (comet) communities. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 271–283. Springer, 2014.

[13] Brett W Bader, Michael W Berry, and Murray Browne. Discussion tracking in enron email using parafac. In *Survey of Text Mining II*, pages 147–163. Springer, 2008.

[14] Brett W Bader, Andrey A Puretskiy, and Michael W Berry. Scenario discovery using nonnegative tensor factorization. In *Iberoamerican Congress on Pattern Recognition*, pages 791–805. Springer, 2008.

[15] James P Bagrow, Dashun Wang, and Albert-Laszlo Barabasi. Collective response of human populations to large-scale emergencies. *PloS one*, 6(3):e17680, 2011.

[16] Yang Bai, Jale Tezcan, Qiang Cheng, and Jie Cheng. A multiway model for predicting earthquake ground motion. In *2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pages 219–224. IEEE, 2013.

[17] Anoop Korattikara Balan, Vivek Rathod, Kevin P Murphy, and Max Welling. Bayesian dark knowledge. In *Advances in Neural Information Processing Systems*, pages 3438–3446, 2015.

[18] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[19] William D Berry, Evan J Ringquist, Richard C Fording, and Russell L Hanson. Measuring citizen and government ideology in the american states, 1960–93. *American Journal of Political Science*, pages 327–348, 1998.

[20] Alex Beutel, Partha Pratim Talukdar, Abhimanu Kumar, Christos Faloutsos, Evangelos E Papalexakis, and Eric P Xing. Flexifact: Scalable flexible factorization of coupled tensors on hadoop. In *SDM*, pages 109–117. SIAM, 2014.

[21] Preeti Bhargava, Thomas Phan, Jiayu Zhou, and Juhan Lee. Who, what, when, and where: Multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In *Proceedings of the 24th international conference on world wide web*, pages 130–140. International World Wide Web Conferences Steering Committee, 2015.

[22] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[23] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM, 2008.

[24] Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.

[25] Yongjie Cai, Hanghang Tong, Wei Fan, Ping Ji, and Qing He. Facets: Fast comprehensive mining of coevolving high-order time series. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 79–88. ACM, 2015.

[26] Nan Cao, Chaoguang Lin, Qiuhan Zhu, Yu-Ru Lin, Xian Teng, and Xidao Wen. Voila: Visual anomaly detection and monitoring with streaming spatiotemporal data. *IEEE transactions on visualization and computer graphics*, 24(1):23–33, 2018.

[27] J Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. *Psychometrika*, 35(3):283–319, 1970.

[28] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[29] Xinyu Chen, Zhaocheng He, and Jiawei Wang. Spatial-temporal traffic speed patterns discovery and incomplete data recovery via svd-combined tensor decomposition. *Transportation Research Part C: Emerging Technologies*, 86:59–77, 2018.

[30] Yi-Lei Chen, Chiou-Ting Hsu, and Hong-Yuan Mark Liao. Simultaneous tensor decomposition and completion using factor priors. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):577–591, 2013.

[31] Zhe Chen, Andrzej Cichocki, and Tomasz M Rutkowski. Constrained non-negative matrix factorization method for eeg analysis in early detection of alzheimer disease. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 5, pages V–V. IEEE, 2006.

[32] Weiyu Cheng, Yanyan Shen, Linpeng Huang, and Yanmin Zhu. Incorporating interpretability into latent factor models via fast influence analysis. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 885–893. ACM, 2019.

[33] Yun Chi and Shenghuo Zhu. Facetcube: a framework of incorporating prior knowledge into non-negative tensor factorization. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 569–578. ACM, 2010.

[34] Joon Hee Choi and S Vishwanathan. Dfacto: Distributed factorization of tensors. In *Proceedings of the 27th International Conference on Neural Information Processing Systems, NIPS'14*, pages 1296–1304, 2014.

[35] Jaegul Choo, Changhyun Lee, Chandan K Reddy, and Haesun Park. Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization. *IEEE transactions on visualization and computer graphics*, 19(12):1992–2001, 2013.

[36] Andrzej Cichocki. Tensor networks for big data analytics and large-scale optimization problems. *arXiv preprint arXiv:1407.3124*, 2014.

[37] Andrzej Cichocki and Anh-Huy Phan. Fast local algorithms for large scale nonnegative matrix and tensor factorizations. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 92(3):708–721, 2009.

[38] Andrzej Cichocki and Rafal Zdunek. Multilayer nonnegative matrix factorisation. *Electronics Letters*, 42(16):1, 2006.

[39] Andrzej Cichocki and Rafal Zdunek. Regularized alternating least squares algorithms for non-negative matrix/tensor factorization. In *International Symposium on Neural Networks*, pages 793–802. Springer, 2007.

[40] Andrzej Cichocki, Rafal Zdunek, and Shun-ichi Amari. Hierarchical als algorithms for nonnegative matrix and 3d tensor factorization. In *International Conference on Independent Component Analysis and Signal Separation*, pages 169–176. Springer, 2007.

[41] Cody A Coleman, Daniel T Seaton, and Isaac Chuang. Probabilistic use cases: Discovering behavioral patterns for predicting certification. In *Proceedings of the Second ACM Conference on Learning at Scale*, pages 141–148. ACM, 2015.

[42] Pierre Comon, Xavier Luciani, and André LF De Almeida. Tensor decompositions, alternating least squares and other tales. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 23(7-8):393–405, 2009.

[43] Justin Cranshaw, Raz Schwartz, Jason Hong, and Norman Sadeh. The livehoods project: Utilizing social media to understand the dynamics of a city. In *Sixth International AAAI Conference on Weblogs and Social Media*, 2012.

[44] Justin Dauwels, Lalit Garg, Arul Earnest, and Leong Khai Pang. Handling missing data in medical questionnaires using tensor decompositions. In *2011 8th International Conference on Information, Communications & Signal Processing*, pages 1–5. IEEE, 2011.

[45] Direction de la Voirie et des dplacements Service des Dplacements. Donnes trafic issues des capteurs permanents. http://opendata.paris.fr/explore/dataset/comptages-routiers-permanents/.

[46] Lieven De Lathauwer and Joséphine Castaing. Tensor-based techniques for the blind separation of ds–cdma signals. *Signal Processing*, 87(2):322–336, 2007.

[47] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[48] Nicholas Diakopoulos, Mor Naaman, and Funda Kivran-Swaine. Diamonds in the rough: Social media visual analytics for journalistic inquiry. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 115–122. IEEE, 2010.

[49] Christopher Donnelly. *Enhancing Personalization Within ASSISTments*. PhD thesis, Worcester Polytechnic Institute, 2015.

[50] Finale Doshi-Velez and Been Kim. Considerations for evaluation and generalization in interpretable machine learning. In *Explainable and Interpretable Models in Computer Vision and Machine Learning*, pages 3–17. Springer, 2018.

[51] Mennatallah El-Assady, Rita Sevastjanova, Fabian Sperrle, Daniel Keim, and Christopher Collins. Progressive learning of topic modeling parameters: a visual analytics framework. *IEEE transactions on visualization and computer graphics*, 24(1):382–391, 2018.

[52] Mennatallah El-Assady, Fabian Sperrle, Oliver Deussen, Daniel Keim, and Christopher Collins. Visual analytics for topic model optimization based on user-steerable speculative execution. *IEEE transactions on visualization and computer graphics*, 25(1):374–384, 2019.

[53] Michael Elad. From exact to approximate solutions. In *Sparse and Redundant Representations*, pages 79–109. Springer, 2010.

[54] Beyza Ermiş, Evrim Acar, and A Taylan Cemgil. Link prediction via generalized coupled tensor factorisation. *arXiv preprint arXiv:1208.6231*, 2012.

[55] Beyza Ermis and A Taylan Cemgil. Liver ct annotation via generalized coupled tensor factorization. *CLEF*, 2014.

[56] Lisette Espín Noboa, Florian Lemmerich, Philipp Singer, and Markus Strohmaier. Discovering and characterizing mobility patterns in urban spaces: A study of manhattan taxi data. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 537–542. International World Wide Web Conferences Steering Committee, 2016.

[57] Zipei Fan, Xuan Song, and Ryosuke Shibasaki. Cityspectrum: a non-negative tensor factorization approach. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2014.

[58] Hadi Fanaee-T and João Gama. Eigenevent: an algorithm for event detection from complex data streams in syndromic surveillance. *Intelligent Data Analysis*, 19(3):597–616, 2015.

[59] Rodrigo Cabral Farias, Jérémy Emile Cohen, Christian Jutten, and Pierre Comon. Joint decompositions with flexible couplings. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 119–126. Springer, 2015.

[60] Mi Fei and Dit-Yan Yeung. Temporal models for predicting student dropout in massive open online courses. In *Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pages 256–263. IEEE, 2015.

[61] Romain Frelat, Martin Lindegren, Tim Spaanheden Denker, Jens Floeter, Heino O Fock, Camilla Sguotti, Moritz Stäbler, Saskia A Otto, and Christian Möllmann. Community ecology in 3d: Tensor decomposition reveals spatio-temporal dynamics of large ecological communities. *PloS one*, 12(11):e0188205, 2017.

[62] Xiao Fu, Kejun Huang, Evangelos E Papalexakis, Hyun-Ah Song, Partha Pratim Talukdar, Nicholas D Sidiropoulos, Christos Faloutsos, and Tom Mitchell. Efficient and distributed algorithms for large-scale generalized canonical correlations analysis. In *Proceedings of the 16th International Conference on Data Mining (ICDM)*, pages 871–876. IEEE, 2016.

[63] Silvia Gandy, Benjamin Recht, and Isao Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2):025010, 2011.

[64] Laetitia Gauvin, André Panisson, and Ciro Cattuto. Detecting the community structure and activity patterns of temporal networks: a non-negative tensor factorization approach. *PloS one*, 9(1):e86028, 2014.

[65] Hancheng Ge, James Caverlee, Nan Zhang, and Anna Squicciarini. Uncovering the spatio-temporal dynamics of memes in the presence of incomplete information. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1493–1502. ACM, 2016.

[66] Paul Gewirtz. On" i know it when i see it". *The Yale Law Journal*, 105(4):1023–1047, 1996.

[67] Nabeel Gillani, Rebecca Eynon, Michael Osborne, Isis Hjorth, and Stephen Roberts. Communication communities in moocs. *arXiv preprint arXiv:1403.4640*, 2014.

[68] Michael Gleicher, Danielle Albers, Rick Walker, Ilir Jusufi, Charles D Hansen, and Jonathan C Roberts. Visual comparison for information visualization. *Information Visualization*, 10(4):289–309, 2011.

[69] Gene H Golub and Christian Reinsch. Singular value decomposition and least squares solutions. In *Linear Algebra*, pages 134–151. Springer, 1971.

[70] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[71] Thomas L Griffiths, Michael I Jordan, Joshua B Tenenbaum, and David M Blei. Hierarchical topic models and the nested chinese restaurant process. In *Proceedings of the 16th International Conference on Neural Information Processing Systems, NIPS'03*, pages 17–24, 2003.

[72] Usama M Fayyad Georges G Grinstein and Andreas Wierse. *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann, 2002.

[73] Julio Guerra, Shaghayegh Sahebi, Yu-Ru Lin, and Peter Brusilovsky. The problem solving genome: Analyzing sequential patterns of student work with parameterized exercises. In *Proceedings of the 7th International Conference on Educational Data Mining, EDM 2014*, pages 153–160, 2014.

[74] Sunil Kumar Gupta, Dinh Phung, Brett Adams, Truyen Tran, and Svetha Venkatesh. Nonnegative shared subspace learning and its application to social media retrieval. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1169–1178. ACM, 2010.

[75] Sunil Kumar Gupta, Dinh Phung, Brett Adams, and Svetha Venkatesh. Regularized nonnegative shared subspace learning. *Data mining and knowledge discovery*, 26(1):57–97, 2013.

[76] Sherif Halawa, Daniel Greene, and John Mitchell. Dropout prediction in moocs using learner activity features. In *Proceedings of the Second European MOOC Stakeholder Summit*, pages 58–65, 2014.

[77] William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*, 2017.

[78] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems*, pages 1135–1143, 2015.

[79] John D Hansen and Justin Reich. Socioeconomic status and mooc enrollment: enriching demographic information with external datasets. In *Proceedings of the Fifth International Conference on Learning Analytics and Knowledge*. ACM, 2015.

[80] RA HARSHMAN. Foundations of the parafac procedure: Models and conditions for an" explanatory" multi-mode factor analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970.

[81] Johan Håstad. Tensor rank is np-complete. *Journal of Algorithms*, 11(4):644–654, 1990.

[82] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182. International World Wide Web Conferences Steering Committee, 2017.

[83] Neil T Heffernan and Cristina Lindquist Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[84] Matthias Heiler and Christoph Schnörr. Controlling sparseness in non-negative tensor factorization. In *European Conference on Computer Vision*, pages 56–67. Springer, 2006.

[85] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[86] Joyce C Ho, Joydeep Ghosh, Steve R Steinhubl, Walter F Stewart, Joshua C Denny, Bradley A Malin, and Jimeng Sun. Limestone: High-throughput candidate phenotype generation via tensor factorization. *Journal of biomedical informatics*, 52:199–211, 2014.

[87] Joyce C Ho, Joydeep Ghosh, and Jimeng Sun. Marble: high-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 115–124. ACM, 2014.

[88] Roya Hosseini, Peter Brusilovsky, Michael Yudelson, and Arto Hellas. Stereotype modeling for problem-solving performance predictions in moocs and traditional courses. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 76–84. ACM, 2017.

[89] Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in neural information processing systems*, pages 1878–1889, 2017.

[90] Chaoran Huang, Lina Yao, Xianzhi Wang, Boualem Benatallah, Shuai Zhang, and Manqing Dong. Expert recommendation via tensor factorization with regularizing hierarchical topical relationships. In *Proceedings of the International Conference on Service-Oriented Computing*, pages 373–387. Springer, 2018.

[91] Kejun Huang, Nicholas D Sidiropoulos, and Athanasios P Liavas. A flexible and efficient algorithmic framework for constrained matrix and tensor factorization. *IEEE Transactions on Signal Processing*, 64(19):5052–5065, 2016.

[92] Masaaki Imaizumi and Kohei Hayashi. Tensor decomposition with smoothness. *ICML2017*, 2017.

[93] Ponpare Japan. Coupon purchase prediction, 2015. data retrieved from Kaggle.com, https://www.kaggle.com/c/coupon-purchase-prediction/data.

[94] Inah Jeon, Evangelos E Papalexakis, U Kang, and Christos Faloutsos. Haten2: Billion-scale tensor decompositions. In *2015 IEEE 31st International Conference on Data Engineering*, pages 1047–1058. IEEE, 2015.

[95] Katy Jordan. Initial trends in enrolment and completion of massive open online courses. *The International Review of Research in Open and Distributed Learning*, 15(1), 2014.

[96] U Kang, Evangelos Papalexakis, Abhay Harpale, and Christos Faloutsos. Gigatensor: scaling tensor analysis up by 100 times-algorithms and discoveries. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 316–324. ACM, 2012.

[97] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 79–86. ACM, 2010.

[98] JG Daniël Karssen and Martijn Wisse. Fall detection in walking robots by multi-way principal component analysis. *Robotica*, 27(2):249–257, 2009.

[99] Rogier Kievit, Willem Eduard Frankenhuis, Lourens Waldorp, and Denny Borsboom. Simpson's paradox in psychological science: a practical guide. *Frontiers in psychology*, 4:513, 2013.

[100] Byung-Hak Kim, Ethan Vizitei, and Varun Ganapathi. Gritnet 2: Real-time student performance prediction with domain adaptation. *arXiv preprint arXiv:1809.06686*, 2018.

[101] Hannah Kim, Jaegul Choo, Jingu Kim, Chandan K Reddy, and Haesun Park. Simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 567–576. ACM, 2015.

[102] Jingu Kim, Yunlong He, and Haesun Park. Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. *Journal of Global Optimization*, 58(2):285–319, 2014.

[103] Minjeong Kim, Kyeongpil Kang, Deokgun Park, Jaegul Choo, and Niklas Elmqvist. Topiclens: Efficient multi-level visual topic exploration of large-scale document collections. *IEEE transactions on visualization and computer graphics*, 23(1):151–160, 2017.

[104] Marius Kloft, Felix Stiehler, Zhilin Zheng, and Niels Pinkwart. Predicting mooc dropout over weeks using machine learning methods. In *Proceedings of the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs*, 2014.

[105] Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[106] Tamara G Kolda and Jimeng Sun. Scalable tensor decompositions for multi-aspect data mining. In *2008 Eighth IEEE international conference on data mining*, pages 363–372. IEEE, 2008.

[107] Tamara Gibson Kolda. Multilinear operators for higher-order decompositions. Technical report, Sandia National Laboratories, 2006.

[108] Danai Koutra, Evangelos E Papalexakis, and Christos Faloutsos. Tensorsplat: Spotting latent anomalies in time. In *Informatics (PCI), 2012 16th Panhellenic Conference on*, pages 144–149. IEEE, 2012.

[109] Daniel Kressner, Michael Steinlechner, and Bart Vandereycken. Low-rank tensor completion by riemannian optimization. *BIT Numerical Mathematics*, 54(2):447–468, 2014.

[110] Pieter M Kroonenberg and Jan De Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45(1):69–97, 1980.

[111] Dana Lahat, Tülay Adali, and Christian Jutten. Multimodal data fusion: an overview of methods, challenges, and prospects. *Proceedings of the IEEE*, 103(9):1449–1477, 2015.

[112] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788, 1999.

[113] Jungwoo Lee, Sejoon Oh, and Lee Sael. Gift: Guided and interpretable factorization for tensors with an application to large-scale multi-platform cancer analysis. *Bioinformatics*, 34(24):4151–4158, 2018.

[114] Ryong Lee and Kazutoshi Sumiya. Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proceedings of the 2nd ACM SIGSPATIAL international workshop on location based social networks*, pages 1–10. ACM, 2010.

[115] SeungChul Lee, Hongbin Liu, MinJeong Kim, Jeong Tai Kim, and ChangKyoo Yoo. Online monitoring and interpretation of periodic diurnal and seasonal variations of

indoor air pollutants in a subway station using parallel factor analysis (parafac). *Energy and Buildings*, 68:87–98, 2014.

[116] Hong Li, Yantao Wei, Luoqing Li, and Yuan Y Tang. Infrared moving target detection and tracking based on tensor locality preserving projection. *infrared physics & technology*, 53(2):77–83, 2010.

[117] Jie Li, Guan Han, Jing Wen, and Xinbo Gao. Robust tensor subspace learning for anomaly detection. *International Journal of Machine Learning and Cybernetics*, 2(2):89–98, 2011.

[118] Wu-Jun Li and Dit-Yan Yeung. Relation regularized matrix factorization. In *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[119] Xutao Li and Michael K Ng. Solving sparse non-negative tensor equations: algorithms and applications. *Frontiers of Mathematics in China*, 10(3):649–680, 2015.

[120] Yu-Ru Lin and Drew Margolin. The ripple of fear, sympathy and solidarity during the boston bombings. *EPJ Data Science*, 3(1):31, 2014.

[121] Yu-Ru Lin, Jimeng Sun, Paul Castro, Ravi Konuru, Hari Sundaram, and Aisling Kelliher. Metafac: community discovery via relational hypergraph factorization. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 527–536. ACM, 2009.

[122] Yu-Ru Lin, Jimeng Sun, Hari Sundaram, Aisling Kelliher, Paul Castro, and Ravi Konuru. Community discovery via metagraph factorization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(3):17, 2011.

[123] Cheng-Yuan Liou, Wei-Chen Cheng, Jiun-Wei Liou, and Daw-Ran Liou. Autoencoder for words. *Neurocomputing*, 139:84–96, 2014.

[124] Cheng-Yuan Liou, Jau-Chi Huang, and Wen-Chie Yang. Modeling word perception using the elman network. *Neurocomputing*, 71(16-18):3150–3157, 2008.

[125] Dongyu Liu, Panpan Xu, and Liu Ren. Tpflow: Progressive partition and multidimensional pattern extraction for large-scale spatio-temporal data analysis. *IEEE transactions on visualization and computer graphics*, 2018.

[126] Ji Liu, Jun Liu, Peter Wonka, and Jieping Ye. Sparse non-negative tensor factorization using columnwise coordinate descent. *Pattern Recognition*, 45(1):649–656, 2012.

[127] Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Tensor completion for estimating missing values in visual data. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):208–220, 2012.

[128] Wei Liu, Jeffrey Chan, James Bailey, Christopher Leckie, and Kotagiri Ramamohanarao. Mining labelled tensors by discovering both their common and discriminative

subspaces. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 614–622. SIAM, 2013.

[129] Stephan Lorenzen, Niklas Hjuler, and Stephen Alstrup. Tracking behavioral patterns among students in an online educational system. In *Proceedings of the 11th International Conference on Educational Data Mining, EDM 2018, Buffalo, NY, USA, July 15-18, 2018*, 2018.

[130] Dijun Luo, Chris Ding, and Heng Huang. Are tensor decomposition solutions unique? on the global convergence hosvd and parafac algorithms. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 148–159. Springer, 2011.

[131] Yong Luo, Dacheng Tao, Kotagiri Ramamohanarao, Chao Xu, and Yonggang Wen. Tensor canonical correlation analysis for multi-view dimension reduction. *IEEE transactions on Knowledge and Data Engineering*, 27(11):3111–3124, 2015.

[132] Michael Madaio, Shang-Tse Chen, Oliver L Haimson, Wenwen Zhang, Xiang Cheng, Matthew Hinds-Aldrich, Duen Horng Chau, and Bistra Dilkina. Firebird: Predicting fire risk and prioritizing fire inspections in atlanta. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 185–194. ACM, 2016.

[133] Takashi Nicholas Maeda, Narushige Shiode, Chen Zhong, Junichiro Mori, and Tetsuo Sakimoto. Detecting and understanding urban changes through decomposing the numbers of visitors arrivals using human mobility data. *Journal of Big Data*, 6(1):4, 2019.

[134] Jorge Maldonado-Mahauad, Mar Pérez-Sanagustín, Pedro Manuel Moreno-Marcos, Carlos Alario-Hoyos, Pedro J Muñoz-Merino, and Carlos Delgado-Kloos. Predicting learners success in a self-paced mooc through sequence patterns of self-regulated learning. In *Proceedings of the European Conference on Technology-Enhanced Learning*, pages 355–369. Springer, 2018.

[135] Hing-Hao Mao, Chung-Jung Wu, Evangelos E Papalexakis, Christos Faloutsos, Kuo-Chen Lee, and Tien-Cheu Kao. Malspot: Multi 2 malicious network behavior patterns analysis. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 1–14. Springer, 2014.

[136] Eduardo Martınez-Montes, Pedro A Valdés-Sosa, Fumikazu Miwakeichi, Robin I Goldman, and Mark S Cohen. Concurrent eeg/fmri analysis by multiway partial least squares. *NeuroImage*, 22(3):1023–1034, 2004.

[137] Koji Maruhashi, Fan Guo, and Christos Faloutsos. Multiaspectforensics: Pattern mining on large-scale heterogeneous networks with tensor analysis. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 203–210. IEEE, 2011.

[138] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158. ACM, 2010.

[139] Charalampos Mavroforakis, Isabel Valera, and Manuel Gomez-Rodriguez. Modeling the dynamics of learning activity on the web. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1421–1430, 2017.

[140] Sharad Mehrotra, Xiaogang Qiu, Zhidong Cao, and Austin Tate. Technological challenges in emergency response. *IEEE Intelligent Systems*, 4:5–8, 2013.

[141] Russell Merris. Laplacian matrices of graphs: a survey. *Linear algebra and its applications*, 197:143–176, 1994.

[142] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 2018.

[143] Fumikazu Miwakeichi, Eduardo Martınez-Montes, Pedro A Valdés-Sosa, Nobuaki Nishiyama, Hiroaki Mizuhara, and Yoko Yamaguchi. Decomposing eeg data into space–time–frequency components using parallel factor analysis. *NeuroImage*, 22(3):1035–1045, 2004.

[144] Shahin Mohammadi, David F Gleich, Tamara G Kolda, and Ananth Grama. Triangular alignment tame: A tensor-based approach for higher-order network alignment. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 14(6):1446–1458, 2017.

[145] Sina Mohseni, Niloofar Zarei, and Eric D Ragan. A survey of evaluation methods and measures for interpretable machine learning. *arXiv preprint arXiv:1811.11839*, 2018.

[146] Morten Mørup, Lars Kai Hansen, and Sidse M Arnfred. Algorithms for sparse nonnegative tucker decompositions. *Neural computation*, 20(8):2112–2131, 2008.

[147] Morten Mørup, Lars Kai Hansen, Josef Parnas, and Sidse M Arnfred. Decomposing the time-frequency representation of eeg using non-negative matrix and multi-way factorization. Technical report, Technical University of Denmark, 2006.

[148] Luis E Mujica, Josep Vehí, Magda Ruiz, Michel Verleysen, Wieslaw Staszewski, and Keith Worden. Multivariate statistics process control for dimensionality reduction in structural assessment. *Mechanical Systems and Signal Processing*, 22(1):155–171, 2008.

[149] Tamara Munzner. A nested model for visualization design and validation. *IEEE transactions on visualization and computer graphics*, 15(6):921–928, 2009.

[150] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Interpretable machine learning: definitions, methods, and applications. *arXiv preprint arXiv:1901.04592*, 2019.

[151] Saurabh Nagrecha, John Z Dillon, and Nitesh V Chawla. Mooc dropout prediction: lessons learned from making pipelines interpretable. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 351–359. International World Wide Web Conferences Steering Committee, 2017.

[152] Atsuhiro Narita, Kohei Hayashi, Ryota Tomioka, and Hisashi Kashima. Tensor factorization using auxiliary information. *Data Mining and Knowledge Discovery*, 25(2):298–324, 2012.

[153] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.

[154] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *ICML*, volume 11, pages 809–816, 2011.

[155] Yue Ning, Sathappan Muthiah, Huzefa Rangwala, and Naren Ramakrishnan. Modeling precursors for event forecasting via nested multi-instance learning. In *Proceedings of the 22th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016.

[156] Paul Nomikos and John F MacGregor. Monitoring batch processes using multiway principal component analysis. *AIChE Journal*, 40(8):1361–1375, 1994.

[157] Paul Nomikos and John F MacGregor. Multi-way partial least squares in monitoring batch processes. *Chemometrics and intelligent laboratory systems*, 30(1):97–108, 1995.

[158] Márcia Oliveira and Joao Gama. Visualization of evolving social networks using actor-level and community-level trajectories. *Expert Systems*, 30(4):306–319, 2013.

[159] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[160] Ivan V Oseledets. Compact matrix form of the d-dimensional tensor decomposition. In *Proceedings of the International Symposium on Nonlinear Theory and its Applications*, 2009.

[161] Ivan V Oseledets and Eugene E Tyrtyshnikov. Breaking the curse of dimensionality, or how to use svd in many dimensions. *SIAM Journal on Scientific Computing*, 31(5):3744–3759, 2009.

[162] Alp Özdemir, Mark A Iwen, and Selin Aviyente. Multiscale tensor decomposition. In *Proceedings of the 2016 50th Asilomar Conference on Signals, Systems and Computers*, pages 625–629. IEEE, 2016.

[163] Alp Özdemir, Mark A Iwen, and Selin Aviyente. Multiscale analysis for higher-order tensors. *arXiv preprint arXiv:1704.08578*, 2017.

[164] Bei Pan, Yu Zheng, David Wilkie, and Cyrus Shahabi. Crowd sensing of traffic anomalies based on human mobility and social media. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 344–353. ACM, 2013.

[165] Linsey Xiaolin Pang, Sanjay Chawla, Wei Liu, and Yu Zheng. On detection of emerging anomalous traffic patterns using gps data. *Data & Knowledge Engineering*, 87:357–373, 2013.

[166] Evangelia Pantraki and Constantine Kotropoulos. Automatic image tagging and recommendation via parafac2. In *Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on*, pages 1–6. IEEE, 2015.

[167] Evangelos Papalexakis and Konstantinos Pelechrinis. thoops: A multi-aspect analytical framework for spatio-temporal basketball data. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 2223–2232. ACM, 2018.

[168] Evangelos Papalexakis, Konstantinos Pelechrinis, and Christos Faloutsos. Spotting misbehaviors in location-based social networks using tensors. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 551–552. ACM, 2014.

[169] Evangelos E Papalexakis, Christos Faloutsos, Tom M Mitchell, Partha Pratim Talukdar, Nicholas D Sidiropoulos, and Brian Murphy. Turbo-smt: Accelerating coupled sparse matrix-tensor factorizations by 200x. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 118–126. SIAM, 2014.

[170] Evangelos E Papalexakis, Christos Faloutsos, and Nicholas D Sidiropoulos. Parcube: Sparse parallelizable tensor decompositions. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 521–536. Springer, 2012.

[171] Evangelos E Papalexakis, Christos Faloutsos, and Nicholas D Sidiropoulos. Tensors for data mining and data fusion: Models, applications, and scalable algorithms. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(2):16, 2017.

[172] Evangelos E Papalexakis, Tom M Mitchell, Nicholas D Sidiropoulos, Christos Faloutsos, Partha Pratim Talukdar, and Brian Murphy. Scoup-smt: Scalable coupled sparse matrix-tensor factorization. *arXiv preprint arXiv:1302.7043*, 2013.

[173] Laura Pappano. The year of the mooc. *The New York Times*, 2(12), 2012.

[174] Judea Pearl et al. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009.

[175] Konstantinos Pelechrinis and Evangelos Papalexakis. thoops: A multi-aspect analytical framework spatio-temporal basketball data using tensor decomposition. *arXiv preprint arXiv:1712.01199*, 2017.

[176] Mark Pelling. Learning from others: the scope and challenges for participatory disaster risk assessment. *Disasters*, 31(4):373–385, 2007.

[177] Wei Peng and Tao Li. Temporal relation co-clustering on directional social network and author-topic evolution. *Knowledge and Information Systems*, 26(3):467–486, 2011.

[178] Anh Huy Phan and Andrzej Cichocki. Tensor decompositions for feature extraction and classification of high dimensional datasets. *Nonlinear theory and its applications, IEICE*, 1(1), 2010.

[179] Miguel A Prada, Janne Toivola, Jyrki Kullaa, and Jaakko HollméN. Three-way analysis of structural health monitoring data. *Neurocomputing*, 80:119–128, 2012.

[180] Jiezhong Qiu, Jie Tang, Tracy Xiao Liu, Jie Gong, Chenhui Zhang, Qian Zhang, and Yufei Xue. Modeling and predicting learning behavior in moocs. In *Proceedings of the 9th ACM international conference on web search and data mining*, 2016.

[181] Stephan Rabanser, Oleksandr Shchur, and Stephan Günnemann. Introduction to tensor decompositions and their applications in machine learning. *arXiv preprint arXiv:1711.10781*, 2017.

[182] Dimitrios Rafailidis and Alexandros Nanopoulos. Modeling the dynamics of user preferences in coupled tensor factorization. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 321–324. ACM, 2014.

[183] Nadine Renard and Salah Bourennane. Improvement of target detection methods by multiway filtering. *IEEE Transactions on Geoscience and Remote Sensing*, 46(8):2407–2417, 2008.

[184] Steffen Rendle. Factorization machines. In *Proceedings of the 10th International Conference on Data Mining (ICDM)*, pages 995–1000. IEEE, 2010.

[185] Steffen Rendle, Leandro Balby Marinho, Alexandros Nanopoulos, and Lars Schmidt-Thieme. Learning optimal ranking with tensor factorization for tag recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 727–736. ACM, 2009.

[186] Giuseppe Ricci, Marco de Gemmis, and Giovanni Semeraro. Mathematical methods of tensor factorization applied to recommender systems. In *New Trends in Databases and Information Systems*, pages 383–388. Springer, 2014.

[187] Shaghayegh Sahebi, Yu-Ru Lin, and Peter Brusilovsky. Tensor factorization for student modeling and performance prediction in unstructured domain. In *Proceedings of the 9th International Conference on Educational Data Mining*, 2016.

[188] John Saint, Dragan Gašević, and Abelardo Pardo. Detecting learning strategies through process mining. In *Proceedings of the European Conference on Technology-Enhanced Learning*, pages 385–398. Springer, 2018.

[189] Anna Sapienza, Alessandro Bessi, and Emilio Ferrara. Non-negative tensor factorization for human behavioral pattern mining in online games. *Information*, 9(3):66, 2018.

[190] Aaron Schein, John Paisley, David M Blei, and Hanna Wallach. Bayesian poisson tensor factorization for inferring multilateral relations from sparse dyadic event counts. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1045–1054. ACM, 2015.

[191] William E Schlenger, Juesta M Caddell, Lori Ebert, B Kathleen Jordan, Kathryn M Rourke, David Wilson, Lisa Thalji, J Michael Dennis, John A Fairbank, and Richard A Kulka. Psychological reactions to terrorist attacks: findings from the national study of americans' reactions to september 11. *Jama*, 288(5):581–588, 2002.

[192] Amnon Shashua and Tamir Hazan. Non-negative tensor factorization with applications to statistics and computer vision. In *Proceedings of the 22nd international conference on Machine learning*, pages 792–799. ACM, 2005.

[193] Xiaoying Shi, Fanshun Lv, Dewen Seng, Baixi Xing, and Jing Chen. Exploring the evolutionary patterns of urban activity areas based on origin-destination data. *IEEE Access*, 7:20416–20431, 2019.

[194] Masamichi Shimosaka, Keisuke Maeda, Takeshi Tsukiji, and Kota Tsubouchi. Forecasting urban dynamics with mobility logs by bilinear poisson regression. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 535–546. ACM, 2015.

[195] Kijung Shin, Lee Sael, and U Kang. Fully scalable methods for distributed tensor factorization. *IEEE Transactions on Knowledge and Data Engineering*, 29(1):100–113, 2016.

[196] Nicholas D Sidiropoulos and Ramakrishna S Budampati. Khatri-rao space-time codes. *IEEE Transactions on Signal Processing*, 50(10):2396–2407, 2002.

[197] Nicholas D Sidiropoulos, Georgios B Giannakis, and Rasmus Bro. Blind parafac receivers for ds-cdma systems. *IEEE Transactions on Signal Processing*, 48(3):810–823, 2000.

[198] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 650–658. ACM, 2008.

[199] Age Smilde, Rasmus Bro, and Paul Geladi. *Multi-way analysis: applications in the chemical sciences*. John Wiley & Sons, 2005.

[200] Shaden Smith and George Karypis. A medium-grained algorithm for sparse tensor factorization. In *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 902–911. IEEE, 2016.

[201] Qingquan Song, Hancheng Ge, James Caverlee, and Xia Hu. Tensor completion algorithms in big data analytics. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(1):6, 2019.

[202] Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, Teerayut Horanont, Satoshi Ueyama, and Ryosuke Shibasaki. Intelligent system for human behavior analysis and reasoning following large-scale disasters. *IEEE Intelligent Systems*, 28(4):35–42, 2013.

[203] Xuan Song, Quanshi Zhang, Yoshihide Sekimoto, Teerayut Horanont, Satoshi Ueyama, and Ryosuke Shibasaki. Modeling and probabilistic reasoning of population evacuation during large-scale disaster. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013.

[204] Christopher H Stock, Alex H Williams, Madhu S Advani, Andrew M Saxe, and Surya Ganguli. Learning dynamics of deep networks admit low-rank tensor descriptions. *Workshop track - ICLR 2018*, 2018.

[205] Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. Spine: Sparse interpretable neural embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[206] Jimeng Sun, Spiros Papadimitriou, and S Yu Philip. Window-based tensor analysis on high-dimensional and multi-aspect streams. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 1076–1080. IEEE, 2006.

[207] Jimeng Sun, Dacheng Tao, and Christos Faloutsos. Beyond streams and graphs: dynamic tensor analysis. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 374–383. ACM, 2006.

[208] Jimeng Sun, Dacheng Tao, Spiros Papadimitriou, Philip S Yu, and Christos Faloutsos. Incremental tensor analysis: Theory and applications. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2(3):11, 2008.

[209] Yanfeng Sun, Junbin Gao, Xia Hong, Bamdev Mishra, and Baocai Yin. Heterogeneous tensor decomposition for clustering via manifold optimization. *IEEE transactions on pattern analysis and machine intelligence*, 38(3):476–489, 2015.

[210] Panagiotis Symeonidis. Matrix and tensor decomposition in recommender systems. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 429–430. ACM, 2016.

[211] Koh Takeuchi, Yoshinobu Kawahara, and Tomoharu Iwata. Structurally regularized non-negative tensor factorization for spatio-temporal pattern discoveries. In *Joint Eu-*

ropean Conference on Machine Learning and Knowledge Discovery in Databases, pages 582–598. Springer, 2017.

[212] Koh Takeuchi, Ryota Tomioka, Katsuhiko Ishiguro, Akisato Kimura, and Hiroshi Sawada. Non-negative multiple tensor factorization. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1199–1204. IEEE, 2013.

[213] Huachun Tan, Bin Cheng, Wuhong Wang, Yu-Jin Zhang, and Bin Ran. Tensor completion via a multi-linear low-n-rank factorization model. *Neurocomputing*, 133:161–169, 2014.

[214] Huachun Tan, Guangdong Feng, Jianshuai Feng, Wuhong Wang, Yu-Jin Zhang, and Feng Li. A tensor-based method for missing traffic data completion. *Transportation Research Part C: Emerging Technologies*, 28:15–27, 2013.

[215] Huachun Tan, Jianshuai Feng, Guangdong Feng, Wuhong Wang, and Yu-Jin Zhang. Traffic volume data outlier recovery via tensor model. *Mathematical Problems in Engineering*, 2013, 2013.

[216] Dacheng Tao, Xuelong Li, Xindong Wu, and Stephen J Maybank. General tensor discriminant analysis and gabor features for gait recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10), 2007.

[217] Lam Tran, Carmeliza Navasca, and Jiebo Luo. Video detection anomaly via low-rank and sparse decompositions. In *2012 Western New York Image Processing Workshop*, pages 17–20. IEEE, 2012.

[218] Nickolay T Trendafilov. Stepwise estimation of common principal components. *Computational Statistics & Data Analysis*, 54(12):3446–3457, 2010.

[219] Théo Trouillon, Christopher R Dance, Éric Gaussier, Johannes Welbl, Sebastian Riedel, and Guillaume Bouchard. Knowledge graph completion via complex tensor factorization. *The Journal of Machine Learning Research*, 18(1):4735–4772, 2017.

[220] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018.

[221] Ledyard R Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[222] Andranik Tumasjan, Timm Oliver Sprenger, Philipp G Sandner, and Isabell M Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. *ICWSM*, 2010.

[223] Raheem A Usman, FB Olorunfemi, GP Awotayo, AM Tunde, and BA Usman. Disaster risk management and social impact assessment: Understanding preparedness, response and recovery in community projects. In *Environmental Change and Sustainability*. InTech, 2013.

[224] Nico Vervliet, Otto Debals, Laurent Sorber, and Lieven De Lathauwer. Breaking the curse of dimensionality using decompositions of incomplete tensors: Tensor-based scientific computing in big data analysis. *IEEE Signal Processing Magazine*, 31(5):71–79, 2014.

[225] Feng Wang and Li Chen. A nonlinear state space model for identifying at-risk students in open online courses. In *Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016, Raleigh, North Carolina, USA, June 29 - July 2, 2016*, pages 527–532, 2016.

[226] Jingyuan Wang, Fei Gao, Peng Cui, Chao Li, and Zhang Xiong. Discovering urban spatio-temporal structure from time-evolving traffic networks. In *Asia-Pacific Web Conference*, pages 93–104. Springer, 2014.

[227] Wei Wang, Han Yu, and Chunyan Miao. Deep model for dropout prediction in moocs. In *Proceedings of the 2nd International Conference on Crowd Science and Engineering*, pages 26–32. ACM, 2017.

[228] Xiaofeng Wang, Matthew S Gerber, and Donald E Brown. Automatic crime prediction using events extracted from twitter posts. In *International conference on social computing, behavioral-cultural modeling, and prediction*, pages 231–238. Springer, 2012.

[229] Yichen Wang, Robert Chen, Joydeep Ghosh, Joshua C Denny, Abel Kho, You Chen, Bradley A Malin, and Jimeng Sun. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1265–1274. ACM, 2015.

[230] Xidao Wen and Yu-Ru Lin. Sensing distress following a terrorist event. In *Social, Cultural, and Behavioral Modeling: 9th International Conference, SBP-BRiMS 2016, Washington, DC, USA, June 28-July 1, 2016, Proceedings 9*, pages 377–388. Springer, 2016.

[231] Xidao Wen, Yu-Ru Lin, and Konstantinos Pelechrinis. Pairfac: Event analytics through discriminant tensor factorization. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, 2016.

[232] Xidao Wen, Yu-Ru Lin, and Konstantinos Pelechrinis. Event analytics via discriminant tensor factorization. *ACM Trans. Knowl. Discov. Data*, 2018.

[233] Alex H Williams, Tony Hyun Kim, Forea Wang, Saurabh Vyas, Stephen I Ryu, Krishna V Shenoy, Mark Schnitzer, Tamara G Kolda, and Surya Ganguli. Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*, 2018.

[234] Jibing Wu, Zhifei Wang, Yahui Wu, Lihua Liu, Su Deng, and Hongbin Huang. A tensor cp decomposition method for clustering heterogeneous information networks via stochastic gradient descent algorithms. *Scientific Programming*, 2017, 2017.

[235] Xian Wu, Baoxu Shi, Yuxiao Dong, Chao Huang, and Nitesh Chawla. Neural tensor factorization. *arXiv preprint arXiv:1802.04416*, 2018.

[236] Wanli Xing and Dongping Du. Dropout prediction in moocs: Using deep learning for personalized intervention. *Journal of Educational Computing Research*, 2018.

[237] Rui Xu and Donald C Wunsch. Survey of clustering algorithms. *IEEE TRANSACTIONS ON NEURAL NETWORKS*, 2005.

[238] Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013.

[239] Zenglin Xu, Feng Yan, and Yuan Qi. Bayesian nonparametric models for multiway data analysis. *IEEE transactions on pattern analysis and machine intelligence*, 37(2):475–487, 2013.

[240] Yuyu Yan, Yubo Tao, Jin Xu, Shuilin Ren, and Hai Lin. Visual analytics of bike-sharing data based on tensor factorization. *Journal of Visualization*, 21(3):495–509, 2018.

[241] Dingqi Yang, Daqing Zhang, and Bingqing Qu. Participatory cultural mapping based on collective behavior data in location-based social networks. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 7(3):30, 2016.

[242] Tatsuya Yokota and Andrzej Cichocki. Linked tucker2 decomposition for flexible multi-block data analysis. In *International Conference on Neural Information Processing*, pages 111–118. Springer, 2014.

[243] Tatsuya Yokota, Andrzej Cichocki, and Yukihiko Yamashita. Linked parafac/cp tensor decomposition and its fast implementation for multi-block tensor analysis. In *Neural Information Processing*, 2012.

[244] Tatsuya Yokota, Rafal Zdunek, Andrzej Cichocki, and Yukihiko Yamashita. Smooth nonnegative matrix and tensor factorizations for robust multi-way data analysis. *Signal Processing*, 113:234–249, 2015.

[245] Tatsuya Yokota, Qibin Zhao, and Andrzej Cichocki. Smooth parafac decomposition for tensor completion. *IEEE Transactions on Signal Processing*, 64(20):5423–5436, 2016.

[246] Kui Yu, Dawei Wang, Wei Ding, Jian Pei, David L Small, Shafiqul Islam, and Xindong Wu. Tornado forecasting with multiple markov boundaries. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2237–2246. ACM, 2015.

[247] Rafał Zdunek. Approximation of feature vectors in nonnegative matrix factorization with gaussian radial basis functions. In *International Conference on Neural Information Processing*, pages 616–623. Springer, 2012.

[248] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Advances in neural information processing systems*, pages 1601–1608, 2005.

[249] Chidong Zhang, Brian E Mapes, and Brian J Soden. Bimodality in tropical water vapour. *Quarterly Journal of the Royal Meteorological Society*, 129(594):2847–2866, 2003.

[250] Daokun Zhang, Jie Yin, Xingquan Zhu, and Chengqi Zhang. Network representation learning: A survey. *IEEE transactions on Big Data*, 2018.

[251] Fuzheng Zhang, Nicholas Jing Yuan, David Wilkie, Yu Zheng, and Xing Xie. Sensing the pulse of urban refueling behavior: A perspective from taxi mobility. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3):37, 2015.

[252] Tiantian Zhang and Bo Yuan. Visualizing mooc user behaviors: A case study on xuetangx. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 89–98. Springer, 2016.

[253] Xiaoqin Zhang, Xingchu Shi, Weiming Hu, Xi Li, and Steve Maybank. Visual tracking via dynamic tensor analysis with mean update. *Neurocomputing*, 74(17):3277–3285, 2011.

[254] Yuchen Zhang, Jason D Lee, and Michael I Jordan. l1-regularized neural networks are improperly learnable in polynomial time. In *International Conference on Machine Learning*, pages 993–1001, 2016.

[255] Zemin Zhang, Gregory Ely, Shuchin Aeron, Ning Hao, and Misha Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-svd. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3842–3849, 2014.

[256] Qibin Zhao, Guoxu Zhou, Tulay Adali, Liqing Zhang, and Andrzej Cichocki. Kernelization of tensor-based models for multiway data analysis: Processing of multidimensional structured data. *IEEE Signal Processing Magazine*, 30(4):137–148, 2013.

[257] Stephan Zheng, Rose Yu, and Yisong Yue. Multi-resolution tensor learning for large-scale spatial data. *arXiv preprint arXiv:1802.06825*, 2018.

[258] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):38, 2014.

[259] Yu Zheng, Tong Liu, Yilun Wang, Yanmin Zhu, Yanchi Liu, and Eric Chang. Diagnosing new york city's noises with ubiquitous data. In *Proceedings of the 2014 ACM*

International Joint Conference on Pervasive and Ubiquitous Computing, pages 715–725. ACM, 2014.

[260] Guoxu Zhou, Andrzej Cichocki, Qibin Zhao, and Shengli Xie. Efficient nonnegative tucker decompositions: Algorithms and uniqueness. *IEEE Transactions on Image Processing*, 24(12):4990–5003, 2015.

[261] Guoxu Zhou, Qibin Zhao, Yu Zhang, and Andrzej Cichocki. Fast nonnegative tensor factorization by using accelerated proximal gradient. In *International Symposium on Neural Networks*, pages 459–468. Springer, 2014.

[262] Huibin Zhou, Dafang Zhang, Kun Xie, and Yuxiang Chen. Spatio-temporal tensor completion for imputing missing internet traffic data. In *2015 ieee 34th international performance computing and communications conference (ipccc)*, pages 1–7. IEEE, 2015.